Search for solar axions using a 7-GridPix IAXO prototype detector at CAST

Dissertation

zur

Erlangung des Doktorgrades (Dr. rer. nat.)

der

Mathematisch-Naturwissenschaftlichen Fakultät

der

Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt von

Sebastian Michael Schmidt

aus

Solingen

Bonn 2024

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Rheinischen Friedrich-Wilhelms-Universität Bonn

1. Gutachter:	Prof. Dr. Klaus Desch
2. Gutachter:	Prof. Dr. Igor García Irastorza
Tag der Promotion:	25.04.2024
Erscheinungsjahr:	2024

Contents

1. Introduction Intro

Life before death.
Strength before weakness.
Journey before destination.

The Stormlight Archive
by Brandon Sanderson

Mathematics has long served as a guiding tool in theoretical physics. Symmetries, mathematical 'beauty' and the notion of 'naturalness' have long been successfully used to predict new phenomena in physics to be verified experimentally later. Although we may never know whether the universe actually cares about our funny intuitions, in the lack of empirical evidence to the contrary we tend to stick to such approaches. Arguably in the infinite space of mathematical avenues physics *could* express itself, this may be considered an application of Occam's razor to physics. In a sense it underlines the intersection between philosophy, mathematics and physics and indicates that the term 'philosophy of nature' is not actually wholly inapplicable to modern physics. And so we arrive at one of our current representations of this in the form of an angle θ in quantum chromodynamics. Measurements seem to indicate this angle is essentially zero. However, we tend to reject the idea that our universe simply *is* such that θ happens to be close to zero. Instead we derive the simplest explanations as to why this might be the case. And who can blame us when the result of accepting $\theta = 0$ by nature would lead us precisely nowhere? Finding an explanation spawns a new hypothetical friend in our zoo of particles, the *axion*. The point of this thesis is to continue the search for this zoo member by way of staring into the core of the Sun. With the help of a very large number of *virtual* photons we will attempt to entice some axions to become real X-rays, directly detectable by us. And if we fail in this quest, we can put our philosopher's hat back on and muse about how little our axion friends want to dance with our virtual photons.

After a short side note about this thesis as a document in chapter 2, we introduce the theoretical foundation of axion physics in chapter 3. From a historical standpoint as to why axions were invented in the first place to the avenues of detection and related, the expected (model dependent) solar axion flux. We will see that the Sun acts as a strong source of axions in the soft X-ray energy range.

This leads to chapter 4, which fully introduces the concept of an axion helioscope as a way to potentially detect axions of a solar origin. A large magnet is used as a solar telescope in an attempt to reconvert solar axions into photons, X-rays. In particular it introduces the CERN Axion Solar Telescope (CAST) as the experiment at the center of this thesis. Its successor, the International AXion Observatory (IAXO) will also be introduced.

With an understanding of possible detection mechanisms for axions, we will focus next on the required hardware to actually measure axions indirectly. That is, via gaseous detectors for X-ray detection thanks to the axion energy spectrum from the Sun. In chapter 5 we cover the relevant physics related to X-ray interactions with matter and gaseous detector physics.

Next we introduce the detector used in this thesis, the 'Septemboard' 7-GridPix detector, in chapter 6. Here we first introduce the concept of a 'Micromegas detector', which our GridPix detectors are a type of. We will motivate the different detector features this 7-GridPix detector has over the single GridPix detector used previously.

From the hardware of the detector the reader may optionally go over to the data acquisition software and the monitoring tools used during the CAST data taking campaign, in appendix A.

With a fully operational detector in mind, we then introduce the software to reconstruct and analyze data taken with this detector in chapter 8. We discuss cluster finding in the GridPix data, calculation of geometric properties of clusters and reconstruction of FADC spectra.

Then it is finally time to show the installation of the Septemboard detector at the CAST experiment in chapter 9, talk about potential issues encountered which affect data quality and summarize the total data taken, which will later be used for a limit calculation.

Data taken at CAST still needs to be processed and further calibrated to be useful, done in chapter 10. This includes mitigating effects of slight detector instabilities and calibrating the data in energy.

At that point we have everything ready to try and filter the entire dataset to the most X-ray like clusters. When applied to the background dataset this yields our irreducible background rate of events that are either real X-rays due to non-axion sources or other type of X-ray like data. Applied to the axion-sensitive solar tracking dataset the same techniques yield a set of axion candidates. The different classification techniques and how all detector features are used for this purpose is explained in chapter 11.

This finally brings us to chapter 12, in which we introduce our method to evaluate our axion-sensitive solar tracking dataset against our background dataset. This is done using a Bayesian extended likelihood approach. We will mainly compute a limit on the axion-electron coupling constant g_{ae}. But secondarily, we will also consider the axion-photon coupling constant $g_{a\gamma}$ and the chameleon-photon coupling β_γ, a separate hypothetical particle. We will compare each obtained limit with the current best limits.

As a concluding outlook we will discuss potential improvements on different levels possible for

future detectors and physics searches in chapter 13. The lessons learned in this thesis will be summarized to give ideas about which aspects should be emphasized more for future data taking campaigns and which techniques might be worthwhile to investigate for possible improvements to the background rate and similar. This will be placed into context for a potential Timepix3 based detector in the future.

We finally summarize the results and conclude in chapter 14.

2. About this thesis Intro

If you are reading this as a printed thesis, it is likely a shortened version of the full document it is part of. To conform to the expectations of a PhD thesis many parts are removed that are irrelevant for the basic presentation of the work done during the thesis.

However, a fellow researcher who wishes to understand all the details, in particular in terms of reproducibility of the results, the full document should be read instead. If possible, a PhD thesis should be a tome of knowledge about the topic that allows the interested reader to absorb as much of the authors knowledge as possible to help with the continuation of the research. Crucially, this includes details about how results were obtained and access to results in numerical formats, which is both commonly lacking in regular PhD theses.

For this reason the idea of an extended thesis was born, found at [1]

`https://phd.vindaar.de`.

There you will not only find a PDF of the thesis you are likely reading right now, but also:

- a PDF version of the extended document,
- an HTML version of the extended document,
- the raw Org document of the thesis [2],
- a link to the repository of the thesis,
- a link to a repository of the entire raw datasets used in the thesis (plus reconstructed data) following the FAIR [227] guiding principles,
- a large amount of additional documents of notes taken during research and development.

The extended document contains many additional subsections tagged 'extended', similar to the 'Intro' tag (see the top right of the page) of this chapter. These either contain personal thoughts about why certain decisions were taken or additional research done that either lead nowhere or simply is not very relevant for the thesis. Maybe most importantly, you will find a subsection for **every single**

[1] The actual thesis will be found there once the thesis has been published. Links to data, figures and other resources are available before though.

[2] The thesis is not written directly in LaTeX, but in Emacs Org mode. From there it is exported to different targets.

plot and table included in this thesis (that was created by me) with the commands or code snippets to produce them! In addition, the linked git repository of the thesis not only contains all the figures included in the thesis as vector graphics, but also a **CSV file for every plot**. [3] Use them freely under the only condition to reference this thesis when you do so.

In essence the idea is to provide a fully reproducible thesis.

Furthermore, every thesis, especially those relying on large pieces of software, contains mistakes, bugs, wrong assumptions and more. Most of these are not known to the researcher, in case of wrong assumptions possibly due to lack of knowledge in a specific topic. Other times shortcomings are known, but left out for convenience. In this thesis I try to be *transparent* about known shortcomings. There are almost certainly bugs in the referenced code that I'm not aware of, bad assumptions about certain things, etc. Where I *am* aware of sketchy choices and mistakes, I will highlight them honestly.

During the work of my PhD I have written and contributed to a large number of free software / open source projects. The TimepixAnalysis [4] project – the code base at the heart of all data reconstruction and analysis for this thesis – is just one more such project. What this means is that I have no intention of abandoning it in the future, even if I leave academia. As long as there is demand for it (for example for a GridPix3 detector used at BabyIAXO), I'll be happy to maintain and extend it as needed. Future reader, don't let "last updated N years ago" scare you! Just open an issue on Github [5].

Note that I'm always available for questions about my work or this thesis either via Matrix (@vindaar:matrix.org) or on other channels (Twitter / Discord / Github @vindaar) or plainly by email phd@vindaar.de. Please do not hesitate to contact me, even if it is several years since this thesis was initially released.

[3] CSV is not a great format (it's not even standardized after all), but it is convenient for such a purpose.

[4] `https://github.com/Vindaar/TimepixAnalysis`

[5] Or whatever is the go-to in the year you currently reside. I'll be sure to update the link on my website if we move host.

3. Theory of axions Theory

Contents

A deep understanding of the axion requires a diverse set of knowledge of different aspects of the theoretical foundations of modern physics. The Standard Model of particle physics (SM), quantum field theory (QFT) and quantum chromodynamics (QCD). The CPT invariance of the SM and related CP violation [1] of the weak force. The common algebraic structure of the weak and strong force via SU(2) and SU(3). Anomalies in QFT, in particular the Adler-Bell-Jackiw anomaly. The structure of the QCD vacuum and the related U(1) problem, which is solved by instantons and the concepts of Goldstone's theorem. To do justice to all these aspects in the context of a non-theory PhD thesis is not possible [2]. As such, the theory part of this thesis will be kept short and instead a focus is placed on referencing useful material for an interested reader, in particular see sec. 3.1.

At low energies the Standard Model can be described by a combination of three different forces, the electromagnetic, the weak and the strong force. These can be represented mathematically by an internal group structure of U(1)×SU(2)×SU(3), respectively. [3] The weak force, represented by SU(2),

[1] C refers to the discrete transformation of charge conjugation and P for parity transformation. Both refer to the idea of studying a physical system with either (or both) of these transformations applied. A CP conserving theory (or system) would behave exactly the same under the combined transformation. The Standard Model is mathematically CPT invariant (T being time reversal). As such, if a system exhibits different behavior under time reversal it implies a violation of CP to achieve a combined CPT invariance.

[2] At least for me.

[3] U(1) refers to the "circle group", i.e. the group that describes rotations on a unit circle (consider a phase shift on the complex plane). The group operation as such can be considered as multiplication by a complex phase. SU(n) is the

has long since been known to exhibit a CP violation [1]. Due to the similar group structure between the weak and the strong force (SU(2) vs. SU(3)) many parallels exist between the mathematical descriptions of the two forces in the Standard Model. In particular, the following term, [4]

$$\mathcal{L}_\theta = \theta \frac{g_s^2}{32\pi^2} G_a^{\mu\nu} \tilde{G}_{a\mu\nu},$$

is allowed under the requirements for a Standard Model conform term (i.e. gauge invariant, Lorentz invariant and so on). Here, θ is an angle, g_s^2 is the strong coupling constant and $G_\alpha^{\mu\nu}$ the gluon field strength tensor. As a matter of fact, this term is even required if the instanton solution to the U(1) problem is considered [110, 165] and is a result of the complex vacuum structure of QCD [1]. This term violates P and T transformations and as a result of CPT symmetry also violates CP . Peculiarly, any effect expected from this CP violation has still not been observed. One such effect is an expected electric dipole moment of the neutron [69, 70, 33]. Such a dipole moment may naively be expected plainly from the fact that the constituent quarks of a neutron are charged after all. However, very stringent limits place an extremely low upper bound on it at [32, 168]

$$d_{\text{NEDM}} \leq 3 \times 10^{-26}\, e\,\text{cm},$$

where e is the electron charge. Nature's deviation from our expectation in this context is coined the strong CP problem of particle physics. One possible solution to the strong CP problem would be a massless up or down quark, in which case the QCD Lagrangian would feature a global U(1) axial shift symmetry, which could shift $\theta \mapsto 0$. Even in the late 1980's this was not entirely ruled out, despite the already understood mass ratio $m_u/m_d = 5/9$ [222], due to 2nd order chiral effects [126]. However, nowadays based on lattice QCD calculations this has been ruled out [63, 80, 94].

While it is possible that our universe is simply one in which the effect of the strong CP violation is suppressed (or even exactly zero) "by chance", Helen Quinn and Roberto Peccei realized in 1977 [163, 164] that this behavior can be explained in the presence of an additional scalar field. Shortly after both Weinberg and Wilczek [221, 226] realized the implication of such an additional field, namely a pseudo Nambu-Goldstone boson. Wilczek named it the axion, after a washing detergent, as it "washes the Standard Model clean" of the strong CP problem. The most straightforward axion model based on the work by Wilczek and Weinberg yields a coupling of the axion to matter that is already excluded, because they associate the spontaneous symmetry breaking with the electroweak scale.

special unitary group, which means the group of unitary matrices of rank n with determinant 1, where the group operation is matrix multiplication of these matrices (for SU(2) the Pauli matrices multiplied by $\frac{i}{2}$ are a possible set of infinitesimal generators for example).

[4] The parameter θ as written in the equation is actually a compound of a pure θ from the QCD θ vacuum and an electroweak contribution. A chiral transformation is required to go to the physical mass eigenstates to diagonalize the quark mass matrix. This adds a term to θ, $\overline{\theta} = \theta + \arg\det M$. We simply drop the bar over $\overline{\theta}$.

Models for an 'invisible axion' manage to unify the solution to the strong CP problem with the current lack of experimental evidence for an axion-like particle. There are two main models for the invisible axion, the KSVZ (Kim-Shifman-Vainshtein-Zakharov) [127, 204] and the DFSZ (Dine-Fischler-Srednicki-Zhitnitskii) [81, 234] models.

For the most comprehensive modern overview of the theory of axions, different models, best bounds on different axion couplings and a general axion reference, make sure to look into the aptly named "landscape of QCD axion models" [79]!

From here we will first say a few more words about the two invisible axion models in sec. 3.2. Then we will look at the implications an effective axion Lagrangian has on axion interactions. We will sketch the derivation of the axion-photon conversion probability in sec. 3.3. This conversion probability leads to a discussion of the expected solar axion fluxes in sec. 3.4. At this point a very succinct interlude will introduce the chameleon particle, sec. 3.5. Finally, we will briefly go over the current relevant bounds on different axion couplings in sec. 3.6.

3.1. Useful reading material optional

The following is a list of materials I would recommend, if someone wishes to understand the theory underlying axions better. The focus here is on things not already mentioned in the rest of the text. Your mileage may vary, of course.

First of all, at least have a short look at the original papers by Roberto Peccei and Helen Quinn [163, 164], as well as the 'responses' by Wilczek and Weinberg [226, 221].

If your interest is in understanding the reasoning behind the origin of the strong CP problem via the QCD vacuum structure and the U(1) problem, it may be worthwhile to look at (some of) the original papers. See Weinberg's paper introducing it [223] and 't Hooft's paper on symmetry breaking via the Adler-Bell-Jackiw anomaly [1] as the next logical step. In [110] 't Hooft then shows how instantons solve the U(1) problem. In 1999 't Hooft also wrote a 'historical review' [109] about renormalization of gauge theories, which touches on the U(1) problem as well. It may be easier to follow (as these things so often are in hindsight).

In the vein of review papers, Roberto Peccei wrote a nice review paper in 2008 [166] covering the connection between the U(1) problem, the QCD vacuum structure, the strong CP problem and axions. Similarly, Jihn Kim (the K in the KSVZ model) wrote another very nice review in 2010 [128]. I would probably recommend to read these two for a good overview. They give enough references to dive deeper, if desired.

On the side of axion detection, you should of course look into the two papers by Pierre Sikivie, introducing all our modern axion detection experiment techniques [208, 207]. Sikivie wrote another review in 2020 [209] covering search methods for invisible axions.

Another great overview, covering both theory and experimental parts of the axion, are Igor Irastorza's notes here: [119]. And there is also the text book 'Axions' [136], which may be worth a look as a summary of different aspects of axion physics.

If you only want to look into a single document covering all aspects of axions, from theory to experimental searches, look no further than the 'landscape of QCD axion models' [79], also mentioned in the main text.

Finally, in my master thesis [192] I attempted to introduce the relevant physics for the axion (and chameleon) to a level satisfactory to me at the time. Whether I succeeded or not, you can be the judge. This may (or may not) be a good introduction if you are a student, for whom courses in QFT are fresh in their mind.

3.2. Invisible axion models and axion couplings

Kim-Shifman-Vainshtein-Zakharov model The so called KSVZ model [127, 204] is the simplest invisible axion model. It adds a scalar field σ and a superheavy quark Q, which σ couples to via a Yukawa coupling. The main problem with the standard axion is that its energy scale is the electroweak scale $v_F \approx 250\,\text{GeV}$ resulting in too strong interactions. The KSVZ model effectively achieves a symmetry breaking scale $f_a \gg v_F$ and thus results in an 'invisible axion'. It contains a tree-level axion-photon coupling $g_{a\gamma}$, but no axion-electron couplings. The latter can be found at one-loop level [210].

Dine-Fischler-Srednicki-Zhitnitskii model The DFZS axion model [81, 234] is another axion model, in which the scalar field σ couples to two Higgs doublet fields, H_u and H_d. It does not require an extra superheavy quark, in this case the coupling of the scalar to the doublets achieves the decoupling of the axion symmetry breaking scale f_a from the electroweak scale v_F. The end result is the same, an 'invisible axion' which is very light and has only light interactions with other matter. However, in contrast to the KSVZ model axion-lepton couplings appear at tree level!

Generalizations Further generalizations of axion models over the KSVZ and DFSZ models are possible, resulting in more flexible couplings. From a practical standpoint of an axion experiment it is usually better to consider axion interactions via effective couplings, as the limiting factor is detecting something rather than determining its properties. In particular because the two models above yield very small expected coupling constants in regions of axion masses easily accessible via laboratory experiments. Therefore, an effective Lagrangian like the following

$$\mathcal{L}_{a,\text{eff}} = \frac{1}{2}\partial^\mu a \partial_\mu a - \frac{1}{2}m_a^2 a^2 - \frac{g_{a\gamma}}{4}\widetilde{F}^{\mu\nu}F_{\mu\nu}a - g_{ae}\frac{\partial_\mu a}{2m_e}\overline{\psi}_e\gamma^5\gamma^\mu\psi_e, \tag{3.1}$$

which contains an axion-photon coupling $g_{a\gamma}$ and an axion-electron coupling g_{ae} is useful for experimental searches. Limits given on one of these parameters can always be converted to the specific couplings of one of the existing models if needed.

In principle other couplings exist, for example the axion-nucleon coupling g_N. For brevity we ignore these as they are not relevant in the context of this thesis. Future helioscopes like IAXO may be able to be sensitive to at least g_N though [147].

3.3. Implications for axion interactions - conversion probability

Starting with the Lagrangian in eq. 3.1 and extending it by the Lagrangian for a free photon A_ν,

$$\mathcal{L} = \mathcal{L}_{a,\text{eff}} + \mathcal{L}_\gamma = \mathcal{L}_{a,\text{eff}} - \frac{1}{4} F_{\mu\nu} F^{\mu\nu},$$

where $F_{\mu\nu} = \partial_\mu A_\nu - \partial_\nu A_\mu$ is the electromagnetic field strength tensor. And noting that the axion interaction term of eq. 3.1 can be rewritten as,

$$\mathcal{L}_{a\gamma} = \frac{1}{4} g_{a\gamma} \tilde{F}^{\mu\nu} F_{\mu\nu} a = -g_{a\gamma} a \vec{E} \cdot \vec{B},$$

we can apply the Euler Lagrange equations to both the axion a and photon A_ν to derive a modified Klein-Gordon equation for the axion,

$$\left(\Box + m_a^2\right) a = \frac{1}{4} g_{a\gamma} F_{\mu\nu} \tilde{F}^{\mu\nu},$$

which has a photon source term. Similarly, for the photon equation of motion we derive the homogeneous Maxwell equations with an axion source term,

$$\partial_\mu F^{\mu\nu} = g_{a\gamma} (\partial_\mu a) \tilde{F}^{\mu\nu}.$$

Without going into too much detail, let's shortly sketch how one derives the axion-photon conversion probability from here.

By choosing a suitable gauge and specifying directions of electric and magnetic fields in a suitable coordinate system, we can then derive the mixing between photon and axion states. For example if the propagation of particles is along the z axis and we fix the two degrees of freedom of A_ν by the Lorenz gauge ($\partial_\mu A^\mu = 0$) and Coulomb gauge ($\vec{\nabla} \cdot \vec{A} = 0$) we can derive a single equation of motion for the axion and A_ν field by starting with a plane wave approach. We obtain

$$\left[(\omega^2 + \partial_z^2)\mathbf{1} - \mathbf{M}\right] \begin{pmatrix} A_\perp(z) \\ A_\parallel(z) \\ 0 \end{pmatrix} = 0,$$

with the parallel and orthogonal polarization of the photon $A_\parallel$ and $A_\perp$, respectively and matrix $\mathbf{M}$:

$$\mathbf{M} = \begin{pmatrix} m_\gamma^2 & 0 & 0 \\ 0 & m_\gamma^2 & -\omega g_{a\gamma} B_T \\ 0 & -\omega g_{a\gamma} B_T & m_a^2 \end{pmatrix} \text{ where } m_\gamma^2 = \omega_p^2.$$

Here effects of quantum electrodynamic (QED) vacuum polarization and other polarization effects are ignored. ω_p refers to the plasma frequency, $\omega_p = 4\pi\alpha n_e/m_e$, where n_e the electron density and m_e the electron mass. The mass m_γ refers to an effective photon mass that can appear in media, ω is the frequency and B_T the transverse magnetic field. The constant magnetic field B_T appears, because we assume for our purpose the magnetic field is constant along z, the propagation direction of the photon.

By recognizing that the orthogonal component $A_\perp$ is decoupled from the other two, the problem reduces to a 2 dimensional equation. Note that a side effect of this decoupling is that photons produced from an incoming axion in a magnetic field are always linearly polarized in the direction of the external magnetic field! Further, this equation can be linearized in the ultra relativistic limit $m_\gamma^2 \ll \omega^2$ to

$$\left[(\omega + i\partial_z)\mathbf{1} - \frac{\mathbf{M}}{2\omega}\right] \begin{pmatrix} A_\parallel(z) \\ a(z) \end{pmatrix} = 0.$$

As the mass matrix $\mathbf{M}$ is non diagonal, the fields $A_\parallel$ and a are interaction eigenstates and not propagation eigenstates. If we wish to compute the axion to photon conversion probability, we need the propagation eigenstates however. Transforming from one to the other is done by a regular rotation matrix $\mathbf{R}$, which diagonalizes $\mathbf{M}/2\omega$. In the basis of the propagation eigenstates the fields $A'_\parallel$ and a' are then decoupled and can be easily solved by a plane wave solution. The fields we can measure in an experiment are those of the interaction eigenstates of course. [5]

The interaction eigenstates after a distance z can therefore be expressed by

$$\begin{pmatrix} A_\parallel(z) \\ a(z) \end{pmatrix} = \mathbf{R}^{-1}\mathbf{M}_{\text{diag}}\mathbf{R} \begin{pmatrix} A_\parallel(0) \\ a(0) \end{pmatrix},$$

[5] For all practical purposes the terms interaction eigenstates and propagation eigenstates (the latter often also called mass eigenstates) are a convenient tool to work with. A field X in the interaction eigenstate corresponds to the field we can actually measure in an experiment. However, if fields interact, say with another field Y, then along their space and time evolution they may mix. Therefore, it is useful (and convenient) to introduce a propagation eigenstate X' in which that new field (which is different from the physical field X!) propagates without any interaction. The nature of the field interactions have been absorbed into the time and space evolution of the field itself – it is a superposition of the X and Y interaction eigenstates. In the simplest case a field X' in the propagation state may just be oscillating between X and Y, for example.

where $\mathbf{M}_{\text{diag}}$ is the diagonalized mass matrix,

$$\begin{pmatrix} e^{-i\lambda_+ z} & 0 \\ 0 & e^{-\lambda_- z} \end{pmatrix},$$

where $\lambda_{+,-}$ are its eigenvalues and coefficients in the exponential of the plane wave solutions of the propagation eigenstate fields $A'_{\parallel}$ and a' given by

$$\lambda_{+,-} = \pm\frac{1}{4\omega}\sqrt{\left(\omega_P^2 - m_a^2\right)^2 + \left(2\omega g_{a\gamma} B_T\right)^2}.$$

Then finally, one can compute the conversion probability by starting from initial conditions where no electromagnetic field is present, $A_{\parallel}(0) = 0, a(0) = 1$. Computing the resulting $A_{\parallel}(z)$ with these conditions yields the expression which needs to be squared for the probability to measure a photon at distance z when starting purely from axions in an external, transverse magnetic field B_T

$$P_{a\mapsto\gamma}(z) = |A_{\parallel}(z)|^2 = \left(\frac{g_{a\gamma} B_T z}{2}\right)^2 \left(\frac{\sin\left(\frac{qz}{2}\right)}{\frac{qz}{2}}\right)^2, \tag{3.2}$$

with $q = \frac{m_\gamma^2 - m_a^2}{2\omega}$ and we dropped additional terms $\propto g_{a\gamma} B_T$ as arguments to sinc $= \sin(x)/x$, because they are extremely small compared to qz for reasonable axion masses, magnetic fields and coupling constants.

If further the coherence condition $qL < \pi$ is such that $qL \ll \pi$, the sinc term approaches 1 and the relevant conversion probability is finally:

$$P_{a\mapsto\gamma,\text{vacuum}} = \left(\frac{g_{a\gamma} B L}{2}\right)^2, \tag{3.3}$$

where we dropped the T suffix for the transverse magnetic field and replaced z by the more apt L for the length of a magnet. This is the case for long magnets and/or low axion masses, which is generally applicable in this thesis. The point at which this condition does not strictly hold anymore is the axion mass at which helioscope experiments start to lose sensitivity.

Note that the above conversion probability is given in natural units, specifically Lorentz-Heaviside units, $c = \hbar = \varepsilon_0 = 1$ (meaning $\alpha = e^2/4\pi \approx 1/137$). Arguments (B, L) need either be converted to natural units as well or the missing factors need to be added. The same equation in SI units is given by:

$$P_{a\mapsto\gamma,\text{vacuum}} = \varepsilon_0 \hbar c^3 \left(\frac{g_{a\gamma} B L}{2}\right)^2.$$

A detailed derivation for the above can be found in [150]. [6] An initial derivation for the first axion

[6] There is a more detailed derivation written by Biljana Lakić and Krešimir Jakovčić available internally in the IAXO collaboration. If you don't have access to it, reach out!

helioscope prototype is found in [43] based on [174]. Sikivie gives expected rates in his groundbreaking papers about axion experiments, [208, 207] but is extremely short on details. Another source in the form of [175] in which G. Raffelt covers a very large number of topics relevant to axion searches.

3.3.1. Effects of a buffer gas

As seen in the conversion probability above, there is a term for an effective photon mass m_γ as part of q. And indeed, q becomes zero if $m_\gamma = m_a$, which means the suppression effect of the sinc term disappears. This is something that can be used to increase the conversion probability inside of a magnet, by filling it with a buffer gas (for example helium), as introduced in [174, 43]. However, one also needs to account for the attenuation effect of the gas on the produced X-rays. As such the derivation above needs to include this as part of the evolution of the field $\vec{A}$ [7]. By doing this and following the rest of the derivation, the conversion probability in the presence of a buffer gas comes out to:

$$P_{a\to\gamma} = \left(\frac{g_{a\gamma}B}{2}\right)^2 \frac{1}{q^2 + \Gamma^2/4}\left[1 + e^{-\Gamma L} - 2e^{-\frac{\Gamma L}{2}}\cos(qL)\right], \tag{3.4}$$

where Γ is the inverse absorption length for photons (or attenuation length), B the transverse magnetic field, L the length of the magnetic field and q the axion-photon momentum transfer given by:

$$q = \left|\frac{m_\gamma^2 - m_a^2}{2E_a}\right|$$

One can easily see that this reduces to the vacuum case we discussed before, where the attenuation length Γ and effective photon mass m_γ are zero.

Filling the magnet with a buffer gas to induce an effective photon mass and thus minimizing q has been done at CAST with He_4 and He_3 fillings, as we will mention in chapter 4.1.

> Note: for a potential buffer gas run in BabyIAXO I did some calculations about the required gas pressure steps and the effect of different possible filling configurations. These are not really relevant for this thesis, but can be found in [8] both as a PDF as well as the original Org file plus the tangled source code.

[7] This can be done easily by treating the buffer gas as a refractive medium with complex refractive index $n_\gamma = 1 - m_\gamma^2/(2\omega^2) - i\Gamma/2\omega$, which then produces attenuation via the Γ attenuation term. This is useful to know as it relates to the X-ray properties as discussed later in sec. 5.1.1 and sec. 5.1.2.

[8] `http://phd.vindaar.de/docs/bufferGasIAXO/v1/index.html` contains the `axionMass.pdf`, `axionMass.nim` and finally `axionMass.org`, the document from which both other files are generated.

3.4. Solar axion flux

The effective Lagrangian as shown in eq. 3.1 allows for multiple different axion interactions, which result in the production of axions in the Sun. For KSVZ-like axion models (models with only $g_{a\gamma}$) the only interaction allowing for axion production in the Sun is the Primakoff effect [9] for axions. For DFSZ models with an axion-electron coupling g_{ae} multiple other production channels are viable.

One of the first papers to look at the implications of the axion in terms of astrophysical phenomena is [153] by K. Mikaelian. G. Raffelt expanded on this later in [176] with calculations for the Compton and Bremsstrahlung production rates for DFSZ axion models. In [177] he further calculates the production rate for the Primakoff effect (and later reviews the physics [175]). J. Redondo combined all production processes in [178] to compute a full solar axion flux based on the axion-electron coupling g_{ae} using numerical calculations of the expected metallicity contents at different points in the Sun. This is done by making use of the opacities for different elements at different pressures and temperatures as tabulated by the 'Opacity Project' [213, 113, 200, 202, 29, 201], based on the values provided by the numerical AGSS09 [25] solar model. All these considered axion production channels are:

- (*P*) Primakoff production via $g_{a\gamma}$ in both KSVZ and DFSZ axion models

$$\gamma + \gamma \mapsto a$$

- (ff) electron ion bremsstrahlung (in radio astronomy low energetic cases with photons are also called free-free radiation)

$$e + Z \longrightarrow e + Z + a$$

- (*ee*) electron electron bremsstrahlung [176],

$$e + e \longrightarrow e + e + a$$

- (fb) electron capture (also called recombination or free-bound electron transitions)

$$e + Z \longrightarrow a + Z^{-}$$

- (*C*) Compton scattering [176]

$$e + \gamma \longrightarrow e + a$$

- (bb) and de-excitation (bound-bound electron transitions) via an axion

$$Z^{*} \longrightarrow Z + a$$

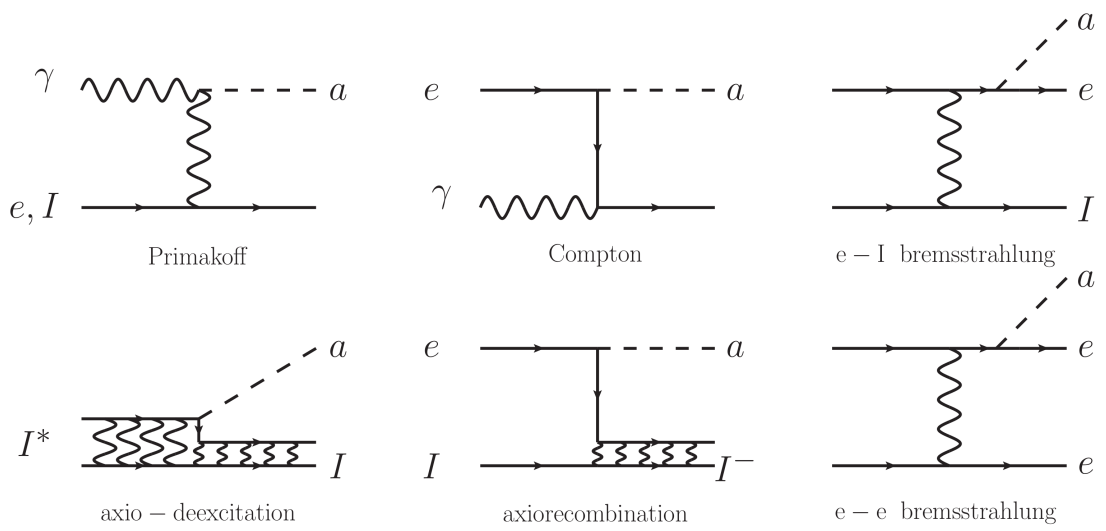


Figure 3.1.: Feynman diagrams of all contributing axion production channels in the Sun for non-hadronic models. In hadronic models only the Primakoff interaction has meaningful contributions, because axion-electron couplings only arise at loop level. Taken from [178].

See fig. 3.1 for the corresponding Feynman diagrams.

With these production channels we can write down the expected axion flux on Earth, based on the production rate per volume in the Sun as the integral

$$\frac{d\Phi_a}{d\omega} = \frac{1}{4\pi R_{\text{Earth}}^2} \int_{\text{Sun}} \mathrm{d}V \frac{4\pi\omega^2}{(2\pi)^3} \Gamma_a(\omega). \tag{3.5}$$

where

$$\Gamma_a(\omega) = \Gamma_a^{\text{ff}} + \Gamma_a^{\text{fb}} + \Gamma_a^{\text{bb}} + \Gamma_a^{C} + \Gamma_a^{ee} + \Gamma_a^{P} \tag{3.6}$$

are all contributing axion production channels. The superscripts correspond to the bullet points above. Most important without going into the details of how each Γ can be expressed, is that they scale with the coupling constant squared. $g_{a\gamma}^2$ for Γ_e^P and g_{ae}^2 for the others. In [178] these production rates are expressed by relating them to the corresponding photon production rates for these processes, which are well known.

For a detailed look, see [178] and the master thesis of Johanna von Oy [158] in which she – among other things – reproduced the calculations done by Redondo. Her work is used as part of this thesis to compute the axion production as required for the expected axion flux in the limit calculation (and provides the data for the plots in this section). The code responsible for computing the emission rates for different axion models is [159], in particular the `readOpacityFile` program. It also uses the Opacity Project as the basis to compute the opacities for different elements.

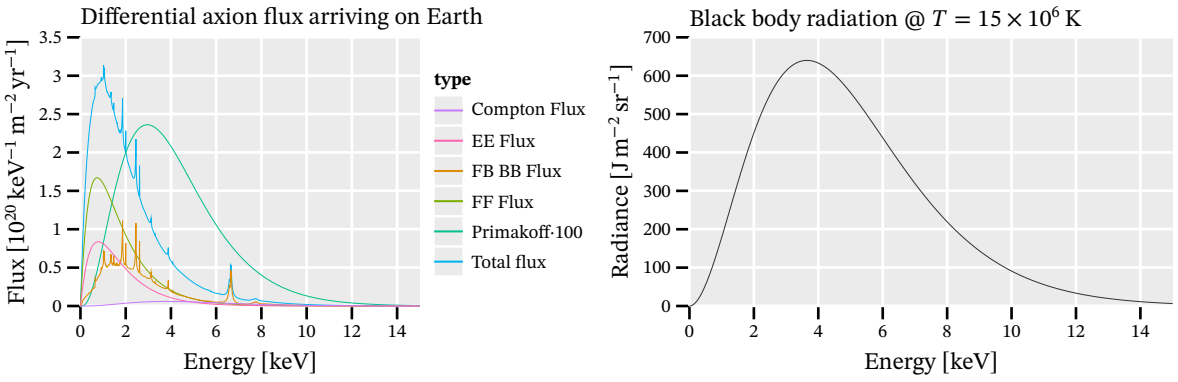
[9] The Primakoff effect – named after Henry Primakoff – is the production of neutral pions in the presence of an atomic nucleus. Due to the pseudoscalar nature and coupling to photons of both neutral pions and axions the equivalent process is allowed for axions.

Fig. 3.2(a) shows the differential axion flux arriving on Earth based on the different contributing interactions, in this case using $g_{ae} = 10^{-13}, g_{a\gamma} = 10^{-12}\,\mathrm{GeV}^{-1}$. We see that the total flux (blue line) peaks at roughly 1 keV, with the small spikes from atomic interactions visible. The Primakoff flux has been amplified by a factor of 100 here to make it more visible, as for this choice of coupling constants its contribution is negligible.

As a result of the expected extremely low mass of the axion, the expected solar axion spectrum for Primakoff only models is essentially a blackbody spectrum corresponding to the temperatures near the core of the Sun, $\mathcal{O}(15 \times 10^6\,\mathrm{K})$, see fig 3.2(b) and compare it to the Primakoff flux on the left [10]. Partially for this reason, in many cases analytical expressions are given to describe the Primakoff axion flux, which are solutions obtained for specific solar models. One such recent result is from [73],

$$\frac{\mathrm{d}\Phi_a}{\mathrm{d}E_a} = \Phi_{P10}\left(\frac{g_{a\gamma}}{10^{-10}\,\mathrm{GeV}^{-1}}\right)^2 \frac{E_a^{2.481}}{e^{E_a/1.205\,\mathrm{keV}}} \tag{3.7}$$

where $\Phi_{P10} = 6.02 \times 10^{10}\,\mathrm{keV}^{-1}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$ and E_a the energy of the axion in keV. [73] also contains analytic expressions for the Compton and Bremsstrahlungs components. For the atomic processes this is not possible.



(a) Differential axion flux

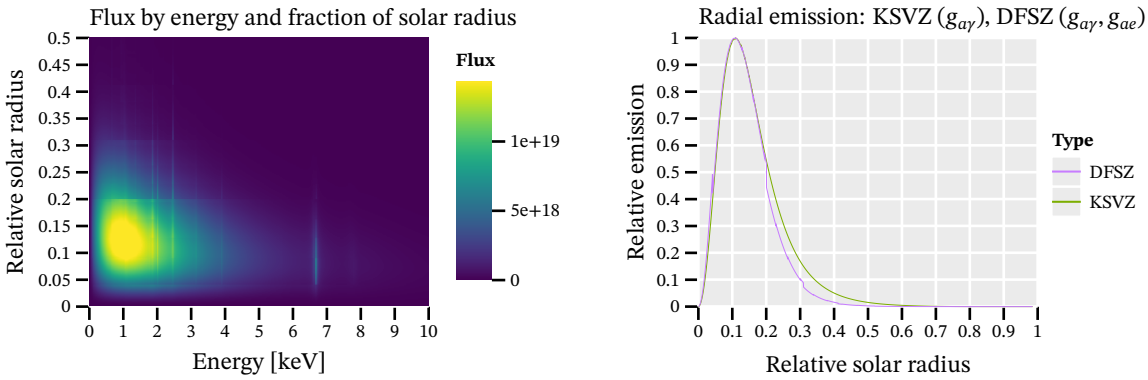
(b) 15×10^6 K blackbody spectrum

Figure 3.2.: (a)Differential solar axion flux based on the different interaction types using $g_{ae} = 10^{-13}, g_{a\gamma} = 10^{-12}\,\mathrm{GeV}^{-1}$. The Primakoff contribution was scaled up by a factor 100 to make it visible, as at these coupling constants the g_{ae} contributions dominate. (b) shows a blackbody spectrum corresponding to 15×10^6 K, roughly the temperature at the solar core. Up to a scaling factor it is essentially the Primakoff flux.

Fig. 3.3(a) shows how the solar axion flux (for DFSZ models) depends both on the energy and the relative radius in the Sun. We can see clearly that the major part of the axion flux comes from a region between 7.5 to 17.5 % of the solar radius. The reason is the cubic scaling of the associated volumes

[10] The difference of the peak position of the blackbody spectrum and the Primakoff flux is due to the axion production being dominantly from 7.5 to 17.5 % of the solar radius (compare fig. 3.3) where temperatures on average are closer to $\sim 12 \times 10^6$ K.

per radius on the lower end and dropping temperatures and densities at the upper end. Interesting substructure due to the details of the axion-electron coupling is visible. The radial component alone comparing it to KSVZ models is seen in fig. 3.3(b), where we can see that the DFSZ flux drops off significantly at a specific radius resulting in the net flux from slightly smaller radii.



(a) Flux (b) Radial emission

Figure 3.3.: (a)Solar axion flux for DFSZ models showing the flux dependence on both the energy and relative solar radius. Dominant contribution below $0.3R_{\odot}$. (b)Difference in the radial contributions of axion flux for KSVZ models against DFSZ models. DFSZ model production is constrained to slightly smaller radii.

The used solar model is an important part of the input for the calculation of the emission rate and thus differential flux. [107] conclude – based on a similar code [108] – that the statistical uncertainty of the solar models is $\mathcal{O}(1\,\%)$, while the systematic uncertainty can reach up to $\mathcal{O}(5\,\%)$. This is an important consideration for the systematic uncertainties later on.

3.5. Chameleons

Chameleons are a different type of hypothetical particle. A scalar particle arising from extensions to general relativity, which acts as a ”fifth force” and can be used to model dark energy. We will not go into detail about the underlying theory here. Refer to [220] for an in-depth introduction to chameleon gravity and [51] on how they differ from other modified gravity models.

Suffice it to say, chameleonic theory also yields a coupling to photons, β_γ, which can be utilized for conversion into X-rays in transverse magnetic fields. Similarly, they can therefore be produced in the Sun. However, the chameleon field has a peculiar property in that its mass is dependent on the density of the surrounding medium. This means chameleons cannot be produced in the solar core and escape. A suitable location to produce chameleons, is the solar tachocline region. A region of differential rotation at the boundary between the inner radiative zone and outer convective zone, at around $0.7 \cdot R_{\odot}$ in the Sun. This leads to strong magnetic fields at low enough densities. Due to the much lower temperatures present in the solar tachocline compared to the solar core, the peak

of the solar chameleon spectrum is below 1 keV. See [52] and [9] for details about the chameleon production in the tachocline region and resulting spectrum.

Their possible detection with CAST was proposed in 2012 [52], with actual searches being performed initially with a Silicon Drift Detector (SDD) [9] and later with a GridPix detector using data taken in 2014/15 [134, 8]. In addition, the KWISP (Kinetic WISP) detector [71, 31] was deployed at CAST, attempting search for the chameleon via their chameleon-matter coupling β_m, which should lead to a chameleons interacting with a force sensor (without conversion to X-rays in the CAST magnet).

As the chameleon flux is highly dependent on details of solar physics that are understood to a much lesser extent than the temperature and composition in the solar core (required for axions), uncertainty on chameleon results is much larger. See also [233] for a discussion of chameleon fields in the context of solar physics.

The conversion probability for back conversion into X-rays in a magnetic field, assuming coherent conversion, is given by [52]:

$$P_{c\mapsto\gamma} = \left(\frac{\beta_\gamma BL}{2m_{\mathrm{pl}}}\right)^2 . \tag{3.8}$$

Here, m_{pl} is the *reduced* Planck mass,

$$m_{\mathrm{pl}} = \frac{M_{\mathrm{pl}}}{\sqrt{8\pi}} = \sqrt{\frac{\hbar c}{8\pi G}},$$

i.e. using natural units with $G = \frac{1}{8\pi}$ instead of $G = 1$, used in cosmology because it removes the 8π term from the Einstein field equations. Similar to the axion-photon conversion probability, it is in natural units and B and L should be converted or the missing constants reinserted. The equation is valid in the chameleon-matter coupling range $1 \leq \beta_m \leq 10^6$, which corresponds to non-resonant production in the Sun as well as evades significant chameleon interaction with materials on its path to the CAST magnet.

Note that in many cases $M_\gamma = \frac{m_{\mathrm{pl}}}{\beta_\gamma}$ is introduced. If however, one defines its inverse, $g_{\beta\gamma} = \frac{\beta_\gamma}{m_{\mathrm{pl}}}$, eq. 3.8 takes the form of eq. 3.3 for the axion, including an effective coupling constant of units GeV^{-1}.

3.6. Current bounds on coupling constants

The field of axion searches is expanding rapidly in recent years, especially in haloscope experiments. A thorough overview of all the different possible ways to constrain axion couplings and best limits in each of them is not in the scope of this thesis. We will give a succinct overview of the general ideas and reference the best current limits on the relevant coupling constants in the regions of interest for

this thesis. A great, frequently updated overview of the current best axion limits is maintained by Ciaran O'Hare, available here [157]. Generally, axion couplings can be probed by three main avenues:

Pure, indirect astrophysical constraints Different astrophysical phenomena can be used to study and constrain axion couplings. One example is the cooling rate of stars. If axions were produced inside of stars and generally manage to leave the star without interaction, they carry energy away. Similar to neutrinos they would therefore contribute to stellar cooling. From observed cooling rates and knowledge of solar models, constraints can be set on potential axion contributions. Many other astrophysical sources can be probed in similar ways. In all cases these constraints are indirect in nature. Which coupling can be constrained depends on the physical processes considered.

Direct astrophysical constraints Certain types of laboratory experiments attempt to measure axions directly and produce constraints due to non-detection. Solar helioscopes attempt to directly measure axions produced in the Sun; more on these in chapter 4. Haloscope experiments utilize microwave cavities in an attempt to tune to the frequency resonant with the axion mass of cold axions part of the dark matter halo. While relying on astrophysically produced axions, the intent is direct detection. Recent interest in axions also means data from WIMP experiments like the XENON collaboration [21] is being analyzed for axion signatures. Haloscopes and helioscopes depend on the axion-photon coupling $g_{a\gamma}$, while WIMP experiments may consider the axion-electron g_{ae} or axion-nucleon g_{aN} couplings. For the astrophysical production mechanism an additional dependency on the coupling producing the axion source is added, which may result in only being able to give constraints on products of different couplings.

Direct production constraints The final approach is full laboratory experiments, which both attempt to first produce axions and then detect them. This idea is commonly done in so called 'light shining through the wall' (LSW) experiments like the ALPS experiment at DESY [30]. Here, a laser cavity in a magnetic field is intended as an axion production facility. These produced axions would leave the cavities, propagate through some kind of wall (e.g. lead) and enter a second set of equivalent cavities, just without an active laser. Produced axions could convert back into photons in the second set of cavities. The disadvantage is that one deals with the $g_{a\gamma}$ coupling both in production and reconversion. [11]

The bounds of interest for this thesis are the axion-photon coupling $g_{a\gamma}$ for masses below around

[11] While astrophysical sources of course also introduce an additional g^2 from their production (resulting in all experiments depending on g^4 effectively), the advantage is that in absolute terms an astrophysical source produces orders of magnitude more axion flux than a LSW experiment. In that sense LSW experiments deal with a squared suppression over those depending on astrophysical sources.

100 meV [12] and the product of the axion-photon and axion-electron couplings $g_{ae} \cdot g_{a\gamma}$ in similar mass ranges. The latter for the case of dominant axion-electron production g_{ae} in the Sun and detection via $g_{a\gamma}$ in the CAST magnet.

The current best limit on the axion-photon coupling from laboratory experiments is from the CAST Nature paper in 2017 [64], providing a bound of

$$g_{a\gamma,\text{Nature}} < 6.6 \times 10^{-11}\,\text{GeV}^{-1} \text{ at } 95\,\% \text{ CL.}$$

Similarly, for the product on the axion-electron and axion-photon couplings, the best limit from a helioscope experiment is also from CAST in 2013 [37]. This limit is

$$g_{ae} \cdot g_{a\gamma} \lesssim 8.1 \times 10^{-23}\,\text{GeV}^{-1} \text{ for } m_a \leq 10\,\text{meV}$$

and acts as the main comparison to the the results presented in this thesis.

From astrophysical processes the brightness of the tip of the red-giant branch (TRGB) stars is the most stringent way to restrict the axion-electron coupling g_{ae} alone. This is because axion production would induce more cooling, which would lead to a larger core mass at helium burning ignition, resulting in brighter TRGB stars. [57] calculate a limit of $g_{ae} = 1.3 \times 10^{-13}$ at 95 % CL. A similar limit is obtained in [211]. [40] compute a comparable limit from the White Dwarf luminosity function. However, for purely astrophysical coupling constraints the strong assumptions needing to be made about the underlying physical processes imply these limits are by themselves not sufficient. In fact there is reason to believe at least some of the current astrohpysical bounds are overestimated. In [76] the authors use a machine learning (ML) model to predict the brightness of TRGB stars to allow for much faster simulations of the parameter space relevant for such bounds. Using Markov Chain Monte Carlo models based on the ML output, they show values up to $g_{ae} = 5 \times 10^{-13}$ are not actually excluded if the full uncertainty of stellar parameters is included. As their calculations only went up to such values, even larger couplings may still be allowed.

Finally, observations from X-ray telescope can also be used for limits on the product of $g_{ae} \cdot g_{a\gamma}$. This has been done in [78] based on data from the Suzaku mission and followed up on by the same authors in [77] using data from the Chandra mission. In the latter they compute an exceptionally strong limit of

$$g_{ae} \cdot g_{a\gamma} < 1.3 \times 10^{-25}\,\text{GeV}^{-1} \text{ at } 95\,\% \text{ CL,}$$

valid for axion masses below $m_a \lesssim 5 \times 10^{-6}$ eV.

[12] 100 meV is the range in which the CAST experiment is not in the fully coherent regime of the sinc term of the conversion probability anymore, resulting in significant sensitivity loss.

Chameleons The current best bound on the chameleon-photon coupling was obtained using a previous GridPix detector with data taken at CAST in 2014/15. The observed limit obtained by C. Krieger in [134, 8] is

$$\beta_{\gamma,\text{Krieger}} = 5.74 \times 10^{10} \text{ at } 95\,\% \text{ CL.}$$

3.6.1. Potential solar axion hints

For completeness a few words about previous results which do not actually provide a limit, but rather show small hints of possible axion signals.

One of the first credible hints of such a solar axion signal comes from the XMM Newton telescope. Seasonal variation in the X-ray flux at a level of 11σ is observed. The explanation provided in [85] is axion reconversion into X-rays in Earth's magnetic field. While criticism exists [183], there is anyhow recent interest in this signal [90], this time due to axion quark nuggets (AQN). An AQN explanation would produce a signal up to 100 keV outside the XMM Newton sensitive range. The authors propose to check archival data of the Nuclear Spectroscopic Telescope Array (NuSTAR) or the Gamma-Ray Burst Monitor of the FERMI telescope for such seasonal variations.

Further, while the CAST Nature result [64] provides the current best limit on the axion-photon coupling, its axion candidate dataset actually shows a signal excess at 3 keV at a 3σ level. Statistical effects or possibly not perfectly accounted for systematic variation resulting in slightly more argon fluorescence during tracking data is more likely.

In 2020 the XENON collaboration announced having seen an excess in their electron recoil counts at energies compatible with a solar axion signal [19]. A possible explanation not requiring solar axions was given as trace amounts of tritium below their sensitivity threshold. This garnered a lot of attention, because while only of 3.4σ significance it was the first hint of a potential solar axion signal published by a large collaboration as such. However, combining the resulting axion coupling with astrophysical results indicates a more likely non axion origin for the signal [26, 148]. With the release the first XENONnT results in 2022 [20] about new physics in which no excess is visible, the old result is ruled out. Add to that, the LUX-ZEPLIN collaboration, a similar xenon filled experiment, recently published their results on new physics. Although not as sensitive as XENONnT, but more sensitive than XENON1T, no excess was observed there either [2].

4. Axion helioscopes Theory

Contents

As discussed in the previous chapter in section 3.4, stars are strong axion factories. In 1983 Pierre Sikivie proposed [208, 207] multiple methods to potentially detect axions, one of these making use of this solar axion production. The fundamental realization is that a transverse magnetic field can act as a means to reconvert axions back into photons, as discussed in sec. 3.3. This allows to design a kind of telescope consisting of a magnet, which tracks the Sun. One expects a small fraction of the axions produced in the Sun to reconvert to photons inside of the magnetic field via the inverse Primakoff effect. These photons will carry the energy of the original particles that produced the axions, namely the energy corresponding to the temperature in the solar core. Some kind of X-ray detector is installed behind the magnet. Ideally, an X-ray telescope can be added to focus any potential X-rays from the full magnet volume onto a smaller area of the detector to massively increase the signal-to-noise ratio.

The first implementation of the helioscope idea was the Rochester-Brookhaven-Florida experiment [43, 139]. It was followed by the SUMICO experiment in Tokyo [155, 117, 116]. The third and latest helioscope is the CERN Axion Solar Telescope (CAST), which we will present in more detail in section 4.1. In the final section we will introduce the next generation of axion helioscopes, the International AXion Observatory (IAXO), section 4.2.

4.1. CERN Axion Solar Telescope (CAST)

The CERN Axion Solar Telescope (CAST) was proposed in 1999 [235] and started data taking in 2003 [236]. Fig. 4.1 shows a panorama view of the experiment in its final year of data taking during some maintenance work.

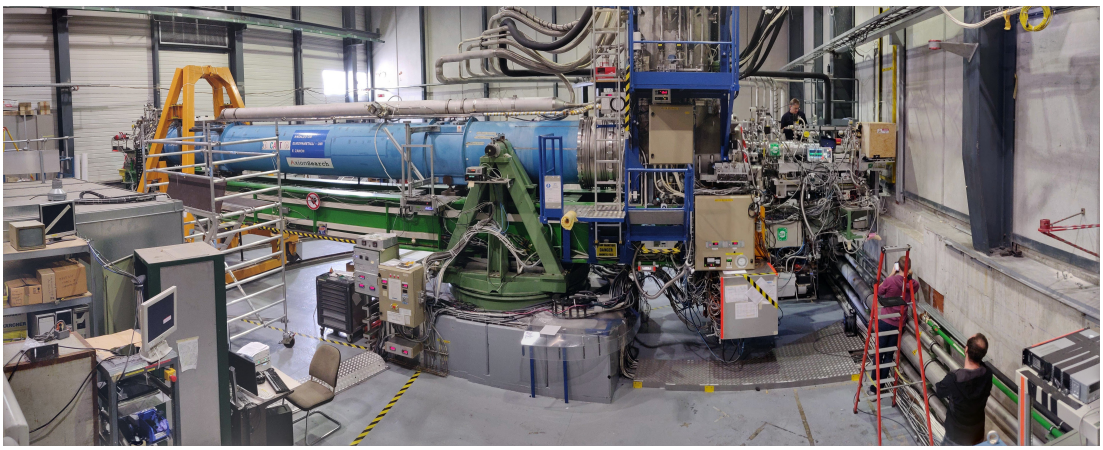


Figure 4.1.: Panorama view of the CAST experiment during some maintenance work.

Using a 9.26 m long Large Hadron Collider (LHC) prototype dipole magnet that was available from the developments for the LHC, CAST features an 8.8 T [1] strong transverse magnetic field for axion-photon conversion produced by a current of 13 kA in the superconducting Nb Ti wires [48] at 1.8 K [47]. It is placed on a movable platform that allows for solar tracking both during sunrise as well as sunset. The vertical range of movement is about $\sim \pm 8\,°$, but practically within ~ -7 to $7.7\,°$ [2]. This range of motion allows for solar tracking of approximately 90 min each during sunrise and sunset per day, the exact duration depending on time of the year. Due to the incredibly feeble interactions of axions, solar tracking can already start before sunrise and stop after sunset as they easily traverse through large distances of Earth's mantle.

An LHC dipole magnet has two bores for the two proton beams running in reverse order. Being a prototype magnet it is **not** bent to the curvature required by the LHC. These two bores have a diameter of 4.3 cm. [3] In total then, two bores on each side allow for 4 experiments to be installed at CAST, two for data taking during sunrise and two during sunset.

The first data taking period (often referred to as 'phase I') took place in 2003 for 6 months between May and November and was a pure vacuum run with 3 different detectors. On the side observing during sunset was a Time Projection Chamber (TPC) that covered both bores. On the 'sunrise' side a Micromegas detector (Micromesh Gaseous Detector) and a Charged Coupled Device (CCD) detector were installed. The CCD was further behind a still in place X-ray telescope originally designed as a

[1] The magnetic field of 8.8 T corresponds to the actual field at which the magnet was operated at CAST, as taken from the magnet slow control data.

[2] These numbers are from CAST's slow control logs. See the extended thesis.

[3] There is some confusion about the diameter and length of the magnet. The original CAST proposal [235] talks about the prototype dipole magnets as having a bore diameter of 42.5 mm and a length of 9.25 m. However, every CAST publication afterwards uses the numbers 43 mm and 9.26 m. Digging into references about the prototype dipole magnets is inconclusive. For better compatibility with all other CAST related publications, we will use the same 43 mm and 9.26 m values in this thesis. Furthermore, measurements were done indicating values around 43 mm.

spare for the ABRIXAS X-ray space telescope [179, 86]. [236]

The full first phase I data taking period comprises of data taken in 2003 and 2004 and achieved a best limit of $g_{a\gamma} < 8.8 \times 10^{-11}\,\text{GeV}^{-1}$ [11].

In what is typically referred to as 'phase II' of the CAST data taking, the magnet was filled with helium as a buffer gas to increase the sensitivity for higher axion masses by inducing an effective photon mass (as mentioned in sec. 3.3.1). First between late 2005 and early 2007 with ^{4}He. From March 2008 a run with ^{3}He was started, which ran until 2011 [22, 23]. 160 steps of different gas pressures were used to optimize sensitivity against time per step. In 2012 another ^{4}He data run took place [23].

From 2013 on the CAST experiment has only run under vacuum configuration [64]. Further, the physics scope has been extended to include searches for chameleons [134, 8, 9, 31, 71] and axions in the galactic halo via cavity experiments [72, 151, 149, 3].

A video showing the CAST magnet during a typical solar tracking can be found under the link in this [4] footnote.

4.1.1. CAST X-ray optics

The first X-ray telescope used at CAST as a focusing optics for the expected axion induced X-ray flux was a Wolter I type X-ray telescope [230] originally built for a proposed German space based X-ray telescope mission, ABRIXAS [179, 86]. The telescope consists of 27 gold coated parabolic and hyperbolic shells and has a focal length of 1.6 m. Due to the small size of the dipole magnet's bores of only 43 mm only a single section of the telescope can be exposed. The telescope is thus placed off-axis from the magnet bore to expose a single mirror section. An image of the mirror system with a rough indication of the exposed section is shown in fig. 4.2(b).

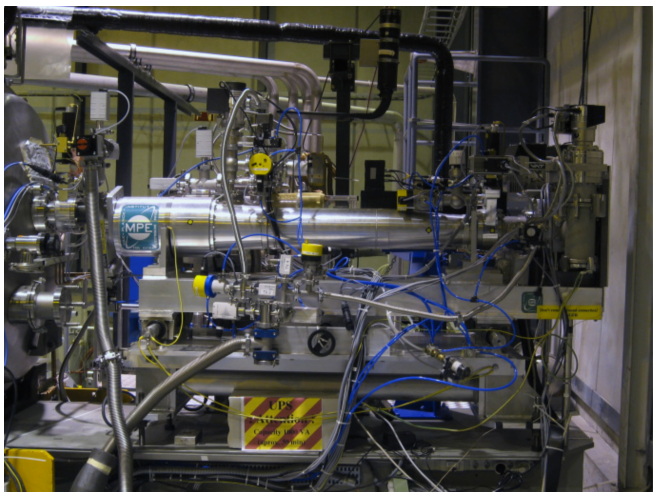
The telescope is owned by the 'Max Planck Institut für extraterrestrische Physik' in Garching. For that reason it will often be referred to as the 'MPE telescope' in the context of CAST.

The efficiency of the telescope reaches about 48 % as the peak at around 1.5 keV, drops sharply at around 2.3 keV to only about 30 % up to about 7 keV. From there it continues to drop until about 5 % efficiency at 10 keV. The efficiency is shown in a comparison with another telescope in the next section 4.1.2 in fig. 4.5.

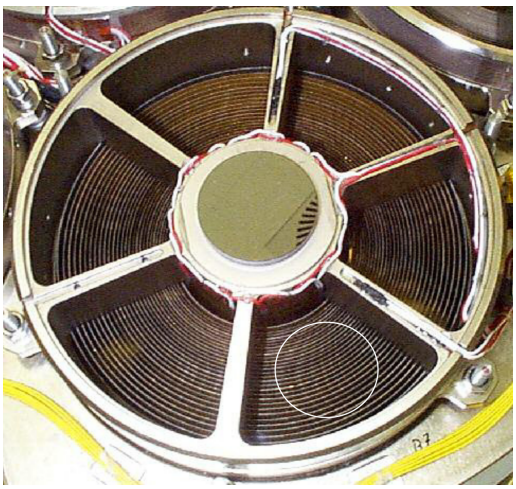
A picture of the telescope installed at CAST behind the magnet on the 'sunrise' side of the magnet is shown in fig. 4.2(a). [5] This telescope was used for the data taking campaign in 2014 and 2015 using a GridPix based detector discussed in [134] and serves as a comparison for certain aspects in this thesis.

[4] https://www.youtube.com/watch?v=XY2lFDXz8aQ

[5] See appendix section D.1 for a schematic of the common terminology like 'sunrise' used at CAST.



(a) Side view



(b) Mirrors

Figure 4.2.: (a): Image of the ABRIXAS telescope installed at CAST on the 'sunrise' side. The image is taken from [135] as it provides a relatively clear image of the telescope, which is hard to take nowadays.(b): Image of the ABRIXAS telescope mirror system. The different shells of the Wolter I type telescope system are visible. One section is exposed to the magnet bore, the white line indicating roughly the extent of the bore. The sproke like structure is the support for the mirror shells. Image taken from [135].

4.1.2. Lawrence Livermore National Laboratory (LLNL) telescope

Up to 2014 there was only a single X-ray telescope in use at CAST. In August 2014 a second X-ray optic was installed on the second bore next to the ABRIXAS telescope. This telescope, using technologies originally developed for the space based NuSTAR telescope by NASA [102, 101, 131, 67, 99], was purpose built for axion searches and in particular the CAST experiment. Contrary to the ABRIXAS telescope only a single telescope section of 30 ° wide mirrors of the telescope was built as the small bore cannot expose more area anyway. It uses a cone approximation to a Wolter I optic, meaning the hyperbolic and parabolic mirrors are replaced by cone sections [169]. A side view and rear view of the open telescope after decommissioning at CAST are seen in fig. 4.3. It consists of 13 platinum / carbon coated glass shells in each telescope section, for a total of 26 mirrors. Each mirror uses one of four different depth graded multilayer (see sec. 5.1.2) coating recipes to improve reflectivity over a larger energy and angle range. Further, the focal length was shortened to 1.5 m. The telescope section is rotated such that the focal point is pointing away from the other magnet bore to make more space for two detectors side by side. [6] This can be seen in the render of the 2017/18 detector setup in fig. 4.4, seen from the top. The development process of the telescope is documented in the PhD thesis by

[6] Because this telescope only consists of one section, its focal point is 'parallel' to the magnet bore. In a full Wolter I like optic the focal point is exactly behind the center of the optic. With only a portion of a telescope this implies an offset. Further, the telescope is not actually rotated exactly 90 ° to move the focal spot perfectly parallel to the two bores, but only about 76 °.

A. Jakobsen [120]. [28] gives an overview and shows preliminary results from CAST. The telescope was characterized and calibrated at the PANTER X-ray test facility in Munich.

(a) Side view (b) Rear view

Figure 4.3.: (a): Image of the LLNL telescope outside its housing after decommissioning at CAST. Entrance side towards the magnet on the right with the largest shell radii. Both sets of mirror shells are clearly seperated in the middle.(b): View of the telescope shells as seen from the detector side. Installed at CAST the telescope was rotated by about 76 ° counter clockwise from this view. Both images courtesy of Cristina Margalejo Blasco.

This telescope achieves a significantly higher effective area than the ABRIXAS telescope in the energy range between 2 to 3 keV fig. 4.5 (relevant for axion searches, compare with fig. 3.2(a)). But outside this range the efficiency is comparable or lower.

The precise understanding of the position, size and shape of the focal point as well as the effective area is essential for the limit calculation later in the thesis. In appendix R we discuss a raytracing simulation of this telescope. The aim is to simulate the expected distribution of axions from the Sun.

4.1.3. Best limits set by CAST

In the many years of data taking and countless detectors taking data at the CAST experiment, it has put the most stringent limits on different coupling constants over the years.

Specifically, CAST sets the current best helioscope limits on the:

- Axion-photon coupling $g_{a\gamma}$
- Axion-electron coupling g_{ae}
- Chameleon-photon coupling β_γ

For the axion-photon coupling the best limit is from [64] in 2017 based on the full Micromegas dataset including the data behind the LLNL telescope and constricts the coupling to $g_{a\gamma} < 6.6 \times 10^{-11}\ \mathrm{GeV}^{-1}$.

For the axion-electron coupling the best limit is still from 2013 in [37] using the theoretical calculations for an expected solar axion flux done by J. Redondo in [178] for a limit on the product of

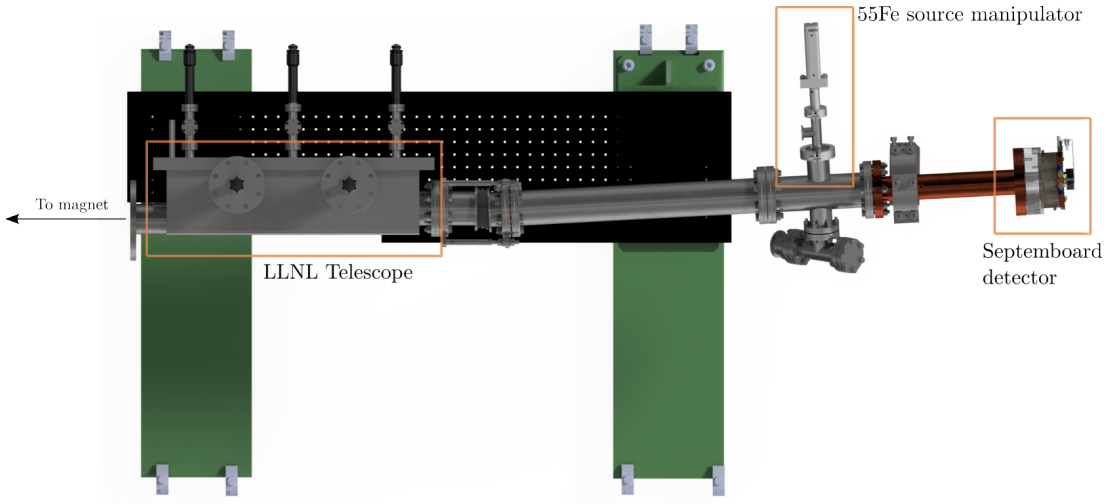


Figure 4.4.: Render of the setup of the GridPix septemboard detector in 2017/18 showing the LLNL telescope on the left side. The diversion away from the extension of the bore is visible, to have more space for detector installation. The lead shielding and veto scintillator are not shown in the render. Render created by Tobias Schiffer.

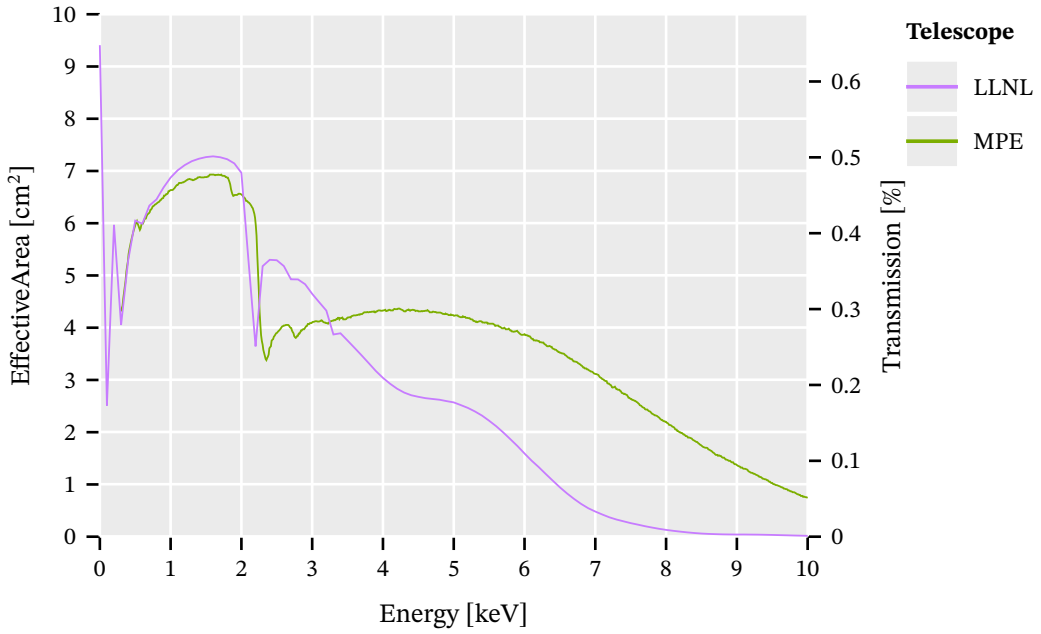


Figure 4.5.: Comparison of the efficiency between the two telescopes, the MPE (ABRIXAS) as the original CAST telescope and the LLNL telescope purpose built for axion searches. The LLNL telescope has superior efficiency in the energy range where the axion flux is assumed to dominate, but falls off sharper at high energies. The data for the LLNL telescope is extracted from fig. 3 in [28], whereas for the ABRIXAS telescope it is extracted from the red line in fig. 4 of [135].

the axion-electron and axion-photon coupling of $g_{ae} \cdot g_{a\gamma} < 8.1 \times 10^{-23}\,\mathrm{GeV}^{-1}$. The limit calculation was based on data taken in CAST phase I in 2003 - 2005 with a pn-CCD detector behind the MPE telescope.

CAST was also used to set a limit on the hypothetical chameleon particle. The best current limit on the chameleon-photon coupling β_γ is based on a single GridPix based detector with data taken in 2014 and 2015 by C. Krieger in [134, 8], limiting the coupling to $\beta_\gamma < 5.74 \times 10^{10}$, which is the first limit below the solar luminosity bound.

4.2. International AXion Observatory (IAXO)

Barring a revolution in detector development or a lucky find of a non QCD axion, the CAST experiment was unlikely to detect any signals. A fourth generation axion helioscope to possibly reach towards the QCD band in the mass-coupling constant phase space is a natural idea.

The first proposal for a next generation axion helioscope was published in 2011 [118], with the name International AXion Observatory (IAXO) first appearing in 2013 [218]. A conceptual design report (CDR) was further published in 2014 [24].

The proposed experiment is supposed to have a total magnet length of 25 m length with eight 60 cm bores with an average transverse magnetic field of 2.5 T [7]. With a cryostat and magnet design specifically built for the experiment, much larger tilting angles of the magnet of about ±25 ° are proposed to allow for solar tracking for 12 h per day for a 1:1 data split between tracking and background data. [24]

In [118] a 'figure of merit' (FOM) is introduced to quantify the improvements possible by IAXO over CAST, defined by

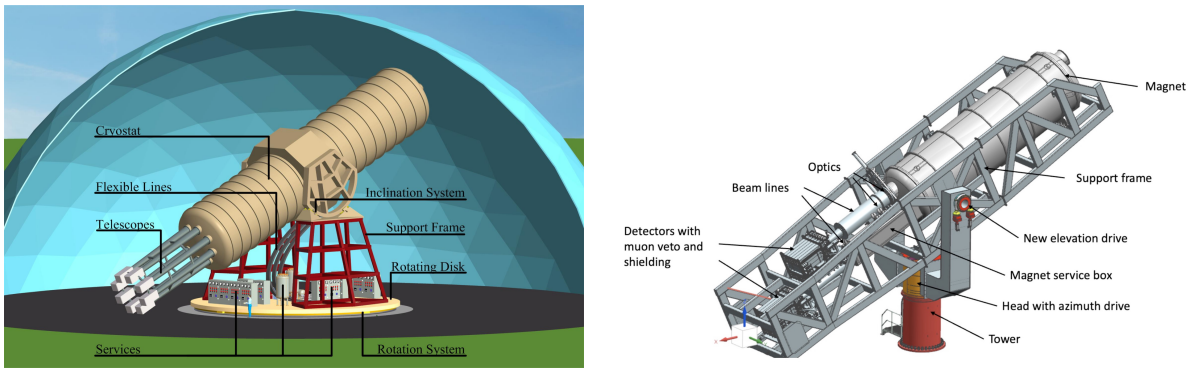
$$f = f_M f_{\mathrm{DO}} f_T = (B^2 L^2 A)_M \left(\frac{\varepsilon_d \varepsilon_o}{\sqrt{ba}}\right)_{\mathrm{DO}} (\sqrt{\varepsilon_t t})_T$$

which is split into individual FOMs for the magnet f_M, the detector and optics f_{DO} and the total tracking time f_T (B: magnetic field, L: magnet length, A: total bore area of all bores, ε_d: detector efficiency, ε_o: X-ray optic efficiency, b: background in counts per area and time, a: area of the X-ray optic focal spot, ε_t: data taking efficiency, t: total solar tracking time). The biggest improvements would come from the magnet FOM (MFOM), due to the much larger magnet volume. Compared to the CAST MFOM $f_{M,\mathrm{CAST}} = 19.3\,\mathrm{T}^2\,\mathrm{m}^4$ IAXO would achieve a relative improvement of $f_{M,\mathrm{CAST}}/f_{M,\mathrm{IAXO}} \approx 300$ with its $f_{M,\mathrm{IAXO}} = 5654.9\,\mathrm{T}^2\,\mathrm{m}^4$. As the number of expected counts in a helio-

[7] Due to the envisioned toroidal magnet design and much larger diameter, the magnetic field inside such a bore would not be as homogeneous as in the LHC dipole magnets. The peak field would be about 5.4 T. This also means a correct f_M calculation for IAXO must take into account the actual magnetic field map.

scope scales with g^4 [8], the figure of merit directly relates possible limits of a helioscope in case of a non-detection. The aspirational target for IAXO would be a full $f_{\mathrm{CAST}}/f_{\mathrm{IAXO}} > 10\,000$ for a possible improvement on g bounds by an order of magnitude.

A schematic of the proposed design can be seen in fig. 4.6(a). Given the comparatively large budget requirements for such an experiment, a compromise was envisioned to prove the required technologies, in particular the magnet design. This intermediate experiment called BabyIAXO will be discussed in the next section, 4.2.1.



(a) IAXO (b) BabyIAXO

Figure 4.6.: (a)An annotated schematic of a potential IAXO design with 8 magnet bores. Taken from the IAXO CDR [24]. (b)An annotated schematic of the current BabyIAXO design, showing the two bores. One bore intended for an XMM-Newton optic of 7.5 m focal length, and a shorter 5 m focal length custom optic behind the other bore. Taken from [180].

4.2.1. BabyIAXO

The major difference between full grown IAXO and BabyIAXO is restricting the setup to 2 bores instead of 8 with a magnet length of only 10 m to prove the magnet design works, before building a larger version of said design. Since the first conceptual design of IAXO [24] the bore diameter for the two bores of BabyIAXO has increased from 60 cm to 70 cm. [65]

The BabyIAXO design was approved by the 'Deutsches Elektronen-Synchrotron' (DESY) for construction onsite. The project has suffered multiple delays most notably due to the magnet construction. First COVID19 due to its severe effects on supply chains and in 2022 the horrific Russian invasion of Ukraine have caused multi year delays. The latter in particular was problematic as the only two companies being able to supply the type of superconducting cable needed for the magnet are from Russia. The magnet situation is still in flow as of writing this thesis.

For the two bores two different X-ray telescopes are planned to be operated. One bore will be

[8] g^4 is a placeholder for $g_{a\gamma}^4$ or $g_{a\gamma}^2 \cdot g_{ae}^2$. The flux scales with this product as explained in sec. 3.6 due to axion production and reconversion happening via g^2 each.

used with a flight spare of the XMM-Newton X-ray satellite mission with a focal length of 7.5 m. The second bore would receive a custom built X-ray optic based on a hybrid design of a NuSTAR-like optic for the inner part – similar to the LLNL telescope introduced in sec. 4.1.2 (just as a full telescope) – and a cold slumped glass design following [62] for the outer part. This optic would have a focal length of only 5 m.

An annotaded schematic of the BabyIAXO design can be seen in fig. 4.6(b). The magnet figure of merit for BabyIAXO is aimed at being at least a factor 10 higher than for CAST, with [65] listing values between 232 to 326 $T^2\,m^4$ depending on how it is calculated.

5. X-rays, cosmic muons and gaseous detectors Theory

Contents

As we have seen in the previous two chapters, solar axions should produce photons in the soft X-ray energy range. In experiments like CAST, gaseous detectors – like the one used in this thesis – are therefore a common and suitable choice for the detection of the X-rays focused by the X-ray optics. In order to give a guiding reference for the relevant physics associated with the detection of particles at CAST, we will now cover each aspect that we make use of in explanations or calculations. The focus will be on aspects either not commonly discussed (for example depth graded multilayers for X-ray telescopes) or essential for explanation (gas diffusion).

We start with a section on particle interactions with matter, sec. 5.1, where we discuss X-ray interactions as well as charged particles in gases. Next in sec. 5.2 follows a short section about cosmic

radiation, in particular the expected muon flux as it serves as a dominant source of background in experiments like CAST. Finally, in sec. 5.3 we cover concepts of gaseous detector physics that we depend on.

5.1. Particle interactions with matter

On the one hand we will discuss how X-rays interact with matter. Both in terms of solids as well as gases, focused on their attenuation, because this is required to describe signal attenuation due to – for example – a detector window of a gaseous detector or the absorption of X-rays in the detector gas. In addition, X-ray reflectivity will be discussed as it is of interest for the behavior of X-ray telescopes. On the other hand the interaction of highly energetic charged particles with matter will be discussed and its relation to cosmic radiation as a source of background. Finally, X-ray fluorescence will be covered as well. As a source of background in an axion helioscope experiment it is indistinguishable [1] from axion-induced X-rays.

For a detailed overview of the interaction of X-rays with matter, see the X-ray data booklet [228].

5.1.1. X-rays in solid matter & gases

Lambert-Beer's law [49, 137, 38]

$$I(z) = I_0 e^{-\mu z}, \tag{5.1}$$

gives the intensity of radiation $I(z)$ after traversing through a medium with constant attenuation μ of length z, given a starting intensity of I_0. Directly related is of course the absorption length $l_{\text{abs}} = 1/\mu$ (or mean free path), which is a useful property when considering typical absorption depths.

This law is of vital importance for the behavior of X-rays traversing through matter, which is needed to compute the efficiency of a gaseous detector with an entrance window. In addition, it is also related to the mean free path of X-rays in a gas, which is an important parameter in gaseous detectors to understand the absorption efficiency of X-rays at different energies and the resulting expected diffusion.

In the context of X-rays the factor μ is typically rewritten via the 'mass attenuation coefficient' $\mu_m = \mu \cdot \rho$ with ρ the density of the material, commonly in $\mathrm{g\,cm^{-3}}$. μ_m is then defined by

$$\mu_m = \frac{N_A}{M}\sigma_A,$$

where N_A is Avogadro's number, M the molar mass of the medium in units of g/mol and σ_A is the photoabsorption cross section in units of $\mathrm{cm^2}$. Thus, the mass attenuation coefficient is usually given

[1] Outside of vetoes to tag muons that cause fluorescence for example.

in $\mathrm{cm}^2\,\mathrm{g}^{-1}$ such that $\mu = \mu_m \cdot \rho$ is of inverse length as expected. This directly yields the definition of the absorption length,

$$l_{\mathrm{abs}} = \frac{1}{\mu}.$$

Further, the photoabsorption cross section can be described via the atomic scattering factor f_2

$$\sigma_A = 2 r_e \lambda f_2,$$

where r_e is the classical electron radius and λ the wavelength of the X-ray. f_2 is the imaginary part of the forward scattering factor f

$$f = f_1 - i f_2$$

which itself is the simplification of the general atomic scattering factor that describes the atom specific part of the scattering cross section.

This way of expressing it has the nice property of relying on a well tabulated parameter f_2. Together with f_1 these tabulated values can be used to compute everything from the refractive index at a specific X-ray energy of a compound to the attenuation coefficient and even reflectivity of a multi-layer substrate. It generalizes from single element to compounds easily by

$$\mu_m = \frac{N_A}{M_c} \sum_i n_i \sigma_{A,i},$$

with M_c the molar weight of the compound and n_i the number of atoms of kind i. X-ray absorption and transmission properties can be calculated from this only requiring the atomic scattering factors, which can be found tabulated for different elements, for example by NIST and Henke.

There is an online calculator for calculations of X-ray transmission found under [2] [105], as well as a library implementation developed during the course of this thesis under [3] [195] for this purpose.

Fig. 5.1 shows an example of X-ray transmission through a 300 nm thick layer of $\mathrm{Si_3\,N_4}$ as well as transmission through 3 cm of argon at normal temperature and pressure (NTP), 1 atm, 20 °C. All information about the absorption lines and general transmission is encoded in f_2.

5.1.2. X-ray reflectivity & scattering

The same atomic scattering factors f_1 and f_2 introduced in section 5.1.1 for the attenuation can also be used to compute the reflectivity of X-rays under shallow angles. A great overview of the relevant

[2] https://henke.lbl.gov/optical_constants/

[3] https://github.com/SciNim/xrayAttenuation

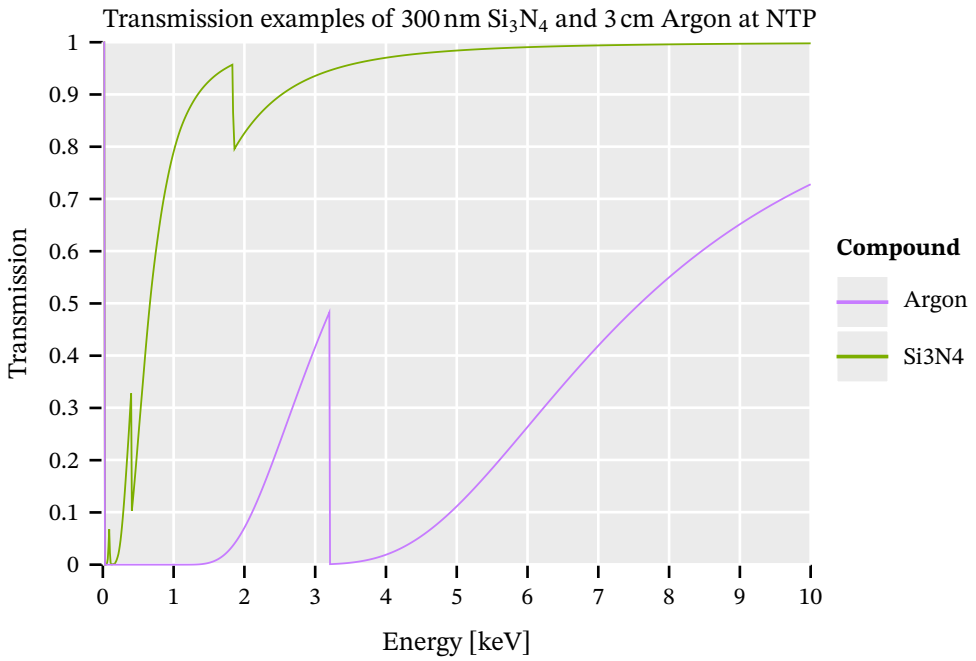


Figure 5.1.: X-ray transmission through a 300 nm thick layer of Si_3N_4 and 3 cm of argon calculated with [195]. Calculation of the transmission based on tabulated scattering form factors.

physics for X-ray reflectivity is found in [229], which introduces the `IMD` program for the simulation of multilayer coatings for X-rays.

By defining the combined scattering factor

$$f(E) = f_1(E) + if_2(E)$$

at energy E, the refractive index n of a medium can be computed using

$$n(E) = 1 - r_e \frac{\lambda^2}{2\pi} \sum_i n_{ai} f_i(E)$$

where n_{ai} is the number density of the i-th compound of the medium. By expressing the refractive index for X-rays in this fashion, reflectivity can be expressed using the Fresnel equations, just like for visible light. The reflectivity for s-polarization is calculated by

$$r_{ik}^s = \frac{n_i \cdot \cos(\theta_i) - n_k \cdot \cos(\theta_k)}{n_i \cdot \cos(\theta_i) + n_k \cdot \cos(\theta_k)} \tag{5.2}$$

while for p-polarization it is done via

$$r_{ik}^p = \frac{n_i \cdot \cos(\theta_k) - n_k \cdot \cos(\theta_i)}{n_i \cdot \cos(\theta_k) + n_k \cdot \cos(\theta_i)}. \tag{5.3}$$

Here θ_i, θ_k are the incident and refracted angles and n_i, n_k the refractive indices on the incident and outgoing side i and k, respectively.

The total reflected energy, the reflectance R, is expressed as

$$R = \frac{1}{2}\left(|r^s|^2 + |r^p|^2\right)$$

for unpolarized light. This can be generalized to multiple layers of material on a substrate and including a surface roughness. Combined, these provide the essence for a realistic computation of the efficiency of an X-ray telescope mirror shell.

This is also implemented in [3] [195] and [2] [105] also provides an online calculator for such reflectivities.

Depth graded multilayers One particular kind of surface, the depth graded multilayer, is used in certain kinds of modern X-ray telescopes, for example the LLNL telescope at CAST following the NuSTAR design. In such a multilayer, repeating layers of a low atomic number Z material and a high Z material are stacked at decreasing thicknesses.

A depth-graded multilayer is described by the equation:

$$d_i = \frac{a}{(b+i)^c}$$

where d_i is the depth of layer i (out of N layers),

$$a = d_{\text{min}}(b+N)^c$$

and

$$b = \frac{1 - Nk}{k-1}$$

with

$$k = \left(\frac{d_{\text{min}}}{d_{\text{max}}}\right)^{\frac{1}{c}}$$

where d_{min} and d_{max} are the thickness of the bottom and top most layers, respectively.

For example for the the LLNL telescope a Pt/C depth graded multilayer is used, in which a top layer of carbon is stacked on top of a platinum layer. Between 2 to 5 repetitions of decreasing thickness are stacked with a ratio of 40 to 45 % carbon to platinum in thickness. More details on this will be discussed in appendix R, as it is of vital importance to calculate the axion image required for the limit calculation correctly.

The reflectivity for a depth-graded multilayer is computed recursively from the bottom of the stack to the top layer using

$$r_i = \frac{r_{ik} + r_k \exp(2i\beta_i)}{1 + r_{ik} r_k \exp(2i\beta_i)}$$

with

$$\beta_i = 2\pi \cdot d_i \cdot \frac{\cos(\theta_i)}{\lambda},$$

where θ_i as seen from the normal axis and with the wavelength of the incoming X-rays λ. The r_{ik} values are computed following equations 5.2 and 5.3. Such multilayers work by summing the reflecting contributions from the different layer transitions. Different thicknesses of the different multilayers mean X-rays at different energies and angles are best reflected from different layers. Thus, a much improved overall reflectivity over a wider energy and angle range can be achieved compared to a normal single layer on a substrate (e.g. a gold coating as used for the XMM-Newton or ABRIXAS optics).

5.1.3. Bethe-Bloch equation

Another relevant aspect for gaseous detectors is the energy deposition of charged particles. In particular for experiments that sit near Earth's surface, a major source of background is due to cosmic radiation, with cosmic muons making up more than 95 % [237] of radiation (aside from neutrinos) at the surface, see sec. 5.2.

The energy loss of such muons can be calculated with the Bethe-Bloch equation, which describes the average energy loss per distance for a charged particle with charge z in a homogeneous medium with charge carriers Z. [237] [4]

$$\left\langle -\frac{\mathrm{d}E}{\mathrm{d}x} \right\rangle = Kz^2 \frac{Z}{A} \frac{1}{\beta^2} \left[\frac{1}{2} \ln \frac{2m_e c^2 \beta^2 \gamma^2 W_{\max}}{I^2} - \beta^2 - \frac{\delta(\beta\gamma)}{2} \right] \tag{5.4}$$

where the different variables are as follows:

- $K = 4\pi N_A r_e^2 m_e c^2 = 0.307\,075\,\mathrm{MeV\,mol^{-1}\,cm^2}$
- $W_{\max}$: maximum possible energy transfer to an electron in a single interaction
- I: mean excitation energy of the absorber material in eV
- $\delta(\beta\gamma)$: density-effect correction to energy loss
- and $r_e = \frac{e^2}{4\pi\varepsilon_0 m_e c^2}$ the classical electron radius, N_A Avogadro's number, m_e electron mass, c speed of light in vacuum, z charge number of incident particle, Z atomic number of absorber material, A atomic mass of absorber material, $\beta = v/c$ speed of incident particle, γ Lorentz factor

This interaction behavior of muons leads to a specific, expected energy loss per distance. Commonly they are 'minimum ionizing particles' (MIPs), as their energies lies between 0.1 to 100 GeV,

[4] Note that there are different common parametrizations of the Bethe-Bloch equation.

the large 'valley' in which the Bethe equation is applicable [5]. For argon gas at normal conditions this is shown in fig. 5.2.

As the Bethe formula was derived from quantum mechanical perturbation theory, higher order corrections can be computed. For our purposes here the leading order is enough. It is important to keep in mind that the Bethe-Bloch equation gives the *mean energy* per distance. When considering short distances as typically encountered in particle detectors, this mean is skewed by rare interactions that deposit large amounts of energy (towards W_{max}). The energy deposition along short distances is typically described by a Landau-Vavilov distribution (similar, but different from a normal Landau distribution) [237, 44]. The most probable energy loss is often a more appropriate number to look at. It can be expressed as

$$\Delta_p = \xi \left[\ln \frac{2m_e c^2 \beta^2 \gamma^2}{I} + \ln \frac{\xi}{I} + j - \beta^2 - \delta(\beta\gamma) \right], \tag{5.5}$$

where ξ is

$$\xi = \frac{1}{2} K z^2 \left\langle \frac{Z}{A} \right\rangle \frac{x}{\beta^2} \,\text{MeV},$$

for a detector in which the material column the particle travels through is expressed as $x = d \cdot \rho$ of a distance d in $\text{g}\,\text{cm}^{-2}$. $j = 0.200$ is an empirical constant [237, 45]. Further, $\langle Z/A \rangle$ is simply the average Z/A for a material compound $\langle Z/A \rangle = \sum_i w_i Z_i / A_i$.

The large difference typically encountered between the most probable and the mean value for the energy loss in particle detectors, makes studying the expected signals a complicated topic. For a detailed description relevant for thin gaseous detectors, see especially [44].

Fig. 5.2 shows the comparison of the most probable energy loss via equation 5.5 and the mean energy loss via the Bethe-Bloch equation 5.4 for muons of different energies traversing 3 cm of argon gas.

5.1.4. X-ray fluorescence

Cosmic muons in their interactions with matter can ionize atoms, leading to the possible emission of X-rays if the removed electron is part of an inner shell, mostly K (and some L) shell electrons. This leads to a form of background based on real X-rays and thus represents a kind of background that is impossible to distinguish from any kind of axion signal unless external scintillator based vetoes are used. Of course, to be relevant as a form of detector background the material must be close to the detector, as the X-rays will otherwise be absorbed. This makes the detector material, the gas

[5] Energies of muons detected near Earth's surface are in this range precisely because they are MIPs. Below and above the much stronger interactions result either in stopping and decay or loss until they are in the MIP range.

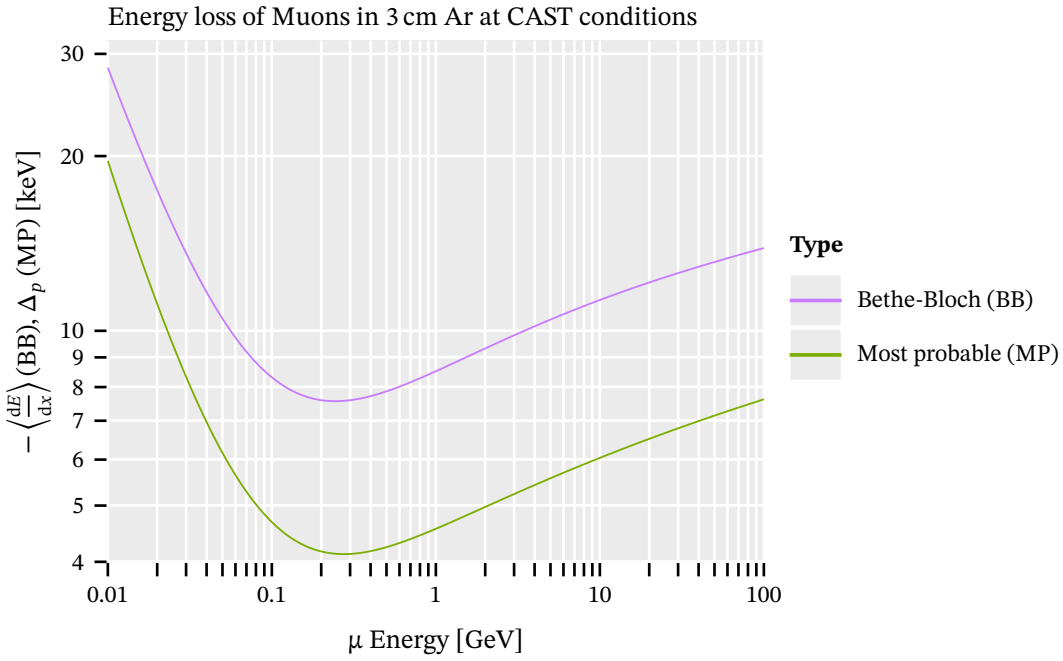


Figure 5.2.: Mean energy loss via Bethe-Bloch (purple) equation of muons in 3 cm of argon at conditions in use in GridPix detector at CAST. 1050 mbar of chamber pressure at room temperature. Note that the mean is skewed by events that transfer a large amount of energy, but are very rare! As such care must be taken interpreting the numbers. Green shows the most probable energy loss, based on the peak of the Landau-Vavilov distribution underlying the Bethe-Bloch mean value.

itself and all material in the direction of the detectors' sensitivity a candidate for X-ray fluorescence background.

Table 5.1 lists the X-ray fluorescence lines for elements one commonly encounters in the context of gaseous detectors for a helioscope experiment. Table 5.2 lists the atomic binding energies of the same elements. This is intended as a useful reference to understand possible lines in background, potential targets for an X-ray tube for reference X-ray data and understand the absorption edges for materials.

The difference between the binding energy and the energies of the most likely fluorescence lines is the explanation for why commonly a material is more transparent for a fluorescence X-ray than energies slightly above it. This (among other effects) explains the 'escape peak' seen in many types of gaseous detectors. For example see fig. 5.1 for the argon transmission. The argon $K1s$ binding energy of 3.2 keV is visible in the form of the sudden drop in transmission (which is of course directly proportional to the absorption length!). At the same time an X-ray produced by an ionized argon atom from the $K1s$ electron via the Kα line yields an X-ray of only 2.95 keV and thus argon gas is extremely transparent for such X-rays (this is the cause of so called 'escape photons', see sec. 5.3.8).

5.2. Cosmic rays

Cosmic rays, or cosmic radiation refers to two aspects of a related phenomenon. Primary cosmic radiation is the radiation arriving at Earth from the Sun, galactic and extragalactic sources. The main contribution are highly energetic protons, but other long lived elementary particles and nuclei also contribute to a lesser extent. Cosmic rays are isotropic at most energies, because of the influence of galactic magnetic fields. Their energies range from 10^9 eV up to 10^{21} eV. It is generally assumed that particles below 10^{18} eV are of mainly galactic origin, whereas the above is dominated by extragalactic sources. The flux of the primary cosmic rays generally follows a power law distribution. Different contributions follow a generally similar power law (also see [237], chapter on cosmic rays).

When cosmic rays interact with the molecules of Earth's atmosphere, mesons are produced, mainly pions. Neutral pions generate showers of photons and electron-positron pairs. Charged pions on the other hand decay into muons and anti muon-neutrinos. Muons are produced over electrons in this case, due to chirality. As they are more massive than electrons they have a larger component of the opposite chirality to the neutrino, which is necessary for this 'forbidden' decay due to angular momentum conservation.

They are produced at an altitude of roughly 15 km. A large fraction of them reaches the surface as they are highly relativistic. Their spectrum is described by a convolution of the production energy, their energy loss due to ionization in the atmosphere and possible decay.

Muons are of interest in the context of helioscope experiments, as they present a dominant source

Z	Element	Kα1	Kα2	Kβ1	Lα1	Lα2	Lβ1	Lβ2	Lγ1	Mα1
6	C	277								
7	N	392.4								
8	O	524.9								
13	Al	1,486.70	1,486.27	1,557.45						
14	Si	1,739.98	1,739.38	1,835.94						
18	Ar	2,957.70	2,955.63	3,190.5						
22	Ti	4,510.84	4,504.86	4,931.81	452.2	452.2	458.4			
25	Mn	5,898.75	5,887.65	6,490.45	637.4	637.4	648.8			
26	Fe	6,403.84	6,390.84	7,057.98	705.0	705.0	718.5			
28	Ni	7,478.15	7,460.89	8,264.66	851.5	851.5	868.8			
29	Cu	8,047.78	8,027.83	8,905.29	929.7	929.7	949.8			
47	Ag	22,162.92	21,990.3	24,942.4	2,984.31	2,978.21	3,150.94	3,347.81	3,519.59	
54	Xe	29,779	29,458	33,624	4,109.9	—	—		—	
78	Pt	66,832	65,112	75,748	9,442.3	9,361.8	11,070.7	11,250.5	12,942.0	2,050.5
79	Au	68,803.7	66,989.5	77,984	9,713.3	9,628.0	11,442.3	11,584.7	13,381.7	2,122.9
82	Pb	74,969.4	72,804.2	84,936	10,551.5	10,449.5	12,613.7	12,622.6	14,764.4	2,345.5

Table 5.1.: Excerpt of X-ray fluorescence energies of K, L and M emission lines for different elements mostly relevant for the context of this thesis in in eV. Taken from the X-ray data book [228], specifically `https://xdb.lbl.gov/Section1/Table_1-2.pdf`.

Z	Element	K 1s	L1 2s	L2 2p1/2	L3 2p3/2	M1 3s	M2 3p1/2	M3 3p3/2	M4 3d3/2	M5 3d5/2	
6	C	284.2									
7	N	409.9	37.3								
8	O	543.1	41.6								
13	Al	1559.6	117.8	72.95	72.55						
14	Si	1839	149.7	99.82	99.42						
18	Ar	3205.9	326.3	250.6	248.4	29.3	15.9	15.7			
22	Ti	4966	560.9	460.2	453.8	58.7	32.6	32.6			
25	Mn	6539	769.1	649.9	638.7	82.3	47.2	47.2			
26	Fe	7112	844.6	719.9	706.8	91.3	52.7	52.7			
28	Ni	8333	1008.6	870.0	852.7	110.8	68.0	66.2			
29	Cu	8979	1096.7	952.3	932.7	122.5	77.3	75.1			
47	Ag	25514	3806	3524	3351	719.0	603.8	573.0	374.0	368.3	
54	Xe	34561	5453	5107	4786	1148.7	1002.1	940.6	689.0	676.4	
78	Pt	78395	13880	13273	11564	3296	3027	2645	2202	2122	
79	Au	80725	14353	13734	11919	3425	3148	2743	2291	2206	
82	Pb	88005	15861	15200	13035	3851	3554	3066	2586	2484	891.8

Z	Element	N1 4s	N2 4p1/2	N3 4p3/2	N4 4d3/2	N5 4d5/2	N6 4f5/2	N7 4f7/2	O1 5s	O2 5p1/2	O3 5p3/2
47	Ag	97.0	63.7	58.3							
54	Xe	213.2	146.7	145.5	69.5	67.5	—	—	23.3	13.4	12.1
78	Pt	725.4	609.1	519.4	331.6	314.6	74.5	71.2	101.7	65.3	51.7
79	Au	762.1	642.7	546.3	353.2	335.1	87.6	84.0	107.2	74.2	57.2
82	Pb	761.9	643.5	434.3	412.2	141.7	136.9	147	106.4	83.3	20.7

Table 5.2.: Excerpt of electron binding energies of elements mostly relevant for the context of this thesis in eV. Taken from the X-ray data book [228], specifically `https://xdb.lbl.gov/Section1/Table_1-1.pdf`.

of background, especially in gaseous detectors (directly and indirectly due to fluorescence). And because current helioscope experiments are built near Earth's surface, little attenuation of muon flux happens. Therefore, a good understanding of the expected muon flux is required.

Above 100 GeV muon decay is negligible. At those energies the muon flux at the surface strictly follows the same power law as the primary cosmic ray flux. Following [88], in this regime it can be described by

$$\frac{\mathrm{d}N_\mu}{\mathrm{d}E_\mu \mathrm{d}\Omega} \approx \frac{0.14 E_\mu^{-2.7}}{\mathrm{cm}^2\,\mathrm{s}\,\mathrm{sr}\,\mathrm{GeV}} \times \left[\frac{1}{1 + \frac{1.1 E_\mu \cos\vartheta}{115\,\mathrm{GeV}}} + \frac{0.054}{1 + \frac{1.1 E_\mu \cos\vartheta}{850\,\mathrm{GeV}}} \right] \tag{5.6}$$

where the first term in parenthesis is the pion and the second the kaon contribution.

For lower energies, [206] provide a set of fitted functions based on 5.6 with a single power law

$$I(E, \theta) = I_0 N (E_0 + E)^{-n} \left(1 + \frac{E}{\varepsilon}\right)^{-1} D(\theta)^{-(n-1)},$$

where I_0, the intensity under zenith angle θ, and ε is another fit parameter for the replacement of the separate meson masses in eq. 5.6. $D(\theta)$ is the path length through the atmosphere under an angle θ from the zenith. N is a normalization constant given by

$$N = (n-1)(E_0 + E_c)^{n-1},$$

where n corresponds to the effective power of the cosine behavior and is the final fit parameter. E_0 accounts for the energy loss due to interactions in the atmosphere and E_c is the lowest energy given in a data set.

If the Earth is assumed flat, it is $D(\theta) = 1/\cos\theta$ (which is often assumed for simplicity and is a reasonable approximation as long as only angles close to $\theta = 0$ are considered). To describe a trajectory through Earth's curved atmosphere however, $D(\theta)$ can be modified to:

$$D(\theta) = \sqrt{\left(\frac{R^2}{d^2}\cos^2\theta + 2\frac{R}{d} + 1\right)} - \frac{R}{d}\cos\theta$$

where R is the Earth radius, d the vertical path length (i.e. the height at which the muon is created) and θ the zenith angle.

While this parametrization is very useful to describe the few specific datasets shown in [206] and provides a way to fit any measured muon flux at a specific location, it is limited in applicability to arbitrary locations, altitudes and angles. For that an approach that does not require a fit to a dataset is preferable, namely by utilizing a combination of the approximation by Gaisser, eq. 5.6, and the interaction of muons with the atmosphere. As such, we modify the equation for the intensity I to the following:

$$I(E,\theta) = I_0(n-1)(E_\theta(E,\theta)+E_c)^{n-1}(E_\theta(E,\theta)+E)^{-n}\left[\frac{1}{1+\frac{1.1E_\mu\cos\vartheta}{115\,\mathrm{GeV}}}+\frac{0.054}{1+\frac{1.1E_\mu\cos\vartheta}{850\,\mathrm{GeV}}}\right]D(\theta)^{-(n-1)},$$

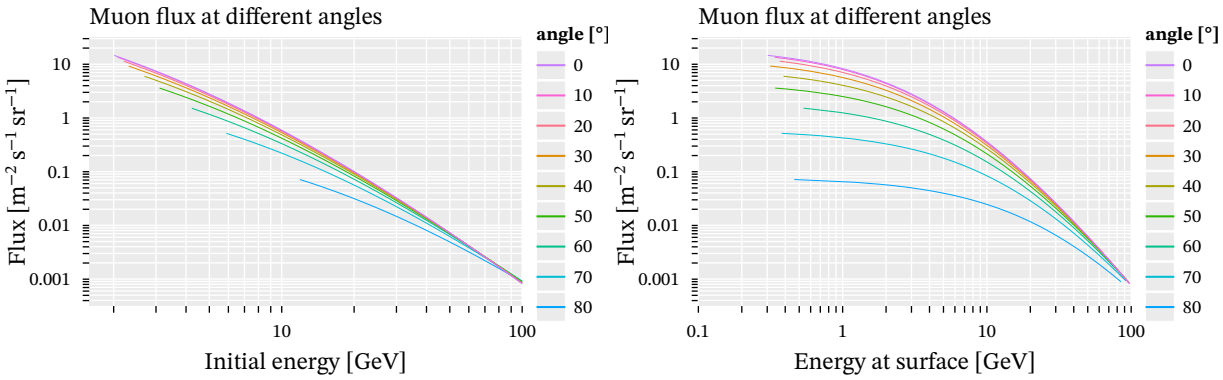
where we take $n = 3$ exactly. One could put in the best fit for the general cosine behavior under zenith angles, $n = n_{\text{fit}} + 1$, but for simplicity we just use 3 here. Take $E_\theta(E,\theta)$ to be the energy of a muon left from initial energy E at generation in the upper atmosphere after transporting it through the atmosphere under the angle θ. The transport must take into account the density change using the barometric height formula of the atmosphere. Transport is done using the Bethe-Bloch equation as introduced in sec. 5.1.3 assuming an atmosphere made up of nitrogen, oxygen and argon. As such we remove all parameters except an initial intensity I_0, which can be set to the best fit of the integrated muon flux at the zenith angle at sea level. In the following figures we simply use $I_0 = 90\,\mathrm{m}^{-2}\,\mathrm{sr}^{-1}\,\mathrm{s}^{-1}$. Figure 5.3 shows the expected differential muon flux using these parameters for different angles at sea level. In 5.3(a) the initial energy of the muons is shown before transporting through the atmosphere. For each angle the lines cut off at the energy below which the muon would likely be stopped by the atmosphere according to its energy loss per distance or reaches the surface with less than 300 MeV. On the right we see the final energy of the same muons at the surface. The lines in 5.3(b) are ragged, because muon decay is simulated using Monte Carlo. [6]

These numbers match reasonably well with different datasets for different locations under different angles, but they should *not* be considered as more than a starting point for a general expectation. However, they are still useful as a reference to consider when evaluating muon fluxes under different angles at CAST.

5.3. Gaseous detector fundamentals

Gaseous detectors consist of a volume filled with gas, usually a noble gas with a small amount of molecular gas. For low rate, low energy experiments an entrance window allows the particles to be detected, to enter. An electric field is applied over the gas volume, strong enough to cause electron-ion pairs created by ionization of the incoming particles to drift to opposite ends of the volume (magnetic fields may be utilized in addition). At least on the side of the anode (where the electrons arrive), a

[6] The calculation is a ”hybrid” Monte Carlo approach. We don’t sample a large number of muons and transport them through the atmosphere. We just compute the loss through the atmosphere and allow the muon to decay randomly, dropping a single data point. This is for simplicity on the one hand and the fact that there is a very sharp transition of ’most muons traverse’ to ’essentially no muons traverse’ the atmosphere at a given energy. For that reason the flux in 5.3(b) at low energies is to be taken with a grain of salt. The fluxes are literally mappings from data points in 5.3(a) to the corresponding energy left at the surface (i.e. assuming no muons decay). Further the 300 MeV energy left is already well inside the energy range where a large number of muons would already decay.



(a) Initial energy

(b) Final energy

Figure 5.3.: Differential muon flux at sea level for different zenith angles. (a) shows the initial energy of the muon. The cutoff corresponds to the lowest energy transported through the atmosphere; muons still arrive at the surface without decay or stopping. (b) shows the final muon energy at the surface, with the lowest muon 300 MeV at the surface.

readout of some form is installed to measure the time of arrival, amount of collected charge and / or the position of the electrons. Depending on the details, this in principle allows for a 3D reconstruction of the initial event in the gas volume. The details of choice of detector gas, applied electric fields, gas volume dimensions and types of readout have a very large impact on the applications a detector is useful for. In the following we will focus on the physics concepts required for gaseous detectors with few cm long drift volumes and high spatial resolution readouts.

Note: this section covers the basic fundamentals that will be later referenced in the thesis. For a much more in-depth understanding of these concepts, see references [186] and [132] and to some extent the PDG [237] (in particular chapters on particle detectors and passage of particles through matter; chapter number varies by year).

5.3.1. Gas mixtures and Dalton's law

Of common interest when dealing with gas mixtures is the notion of partial pressures. In ideal gas theory, a mixture of gases at a pressure P can be considered to be the sum of the 'partial pressures' of each gas species

$$P = \sum_i p_i,$$

initially noted by John Dalton in 1802 [74].

The contribution of each gas only depends on the species' mole fraction. Typically when considering gas mixtures the fractions of each gas species is given as a percentage. The percentage already refers to the mole fraction of each species. As such, the partial pressure of a single species can be

expressed as:

$$p_i = n_i P$$

where n_i is the mole fraction of species i.

This is an extremely valuable property when computing different interactions of particles with gas mixtures. For example when computing the absorption of X-rays after propagating through a certain distance of a gas mixture, as one can compute absorption for each partial pressure independently and combine the contributions after (whether they be added, multiplied or something else of course depends on the considered process).

5.3.2. Ionization energy and average energy per electron-ion pair

In order to understand the signals detected by a gaseous detector, the average number of electrons liberated by an incident particle should be known. This can be calculated given the energy loss in a gas required to produce a single electron-ion pair, called the W-value,

$$W = \frac{I}{\langle N \rangle}.$$

Here, I is the mean ionization energy of the gas and $\langle N \rangle$ the average number of electron-ion pairs generated per interaction. $\langle N \rangle$ is usually smaller than one, 0.6 to 0.7 for noble gases and even below 0.5 for molecular gases. Not all energy of the incoming particle is always deposited in the form of generation of, for example, a photoelectron. Other forms of energy loss are possible. In molecular gases vibrational and rotational modes offer even more possibilities, resulting in the smaller values.

The mean excitation energy I of an element is the weighted combination of the most likely energy levels the element is ionized from. The exact values are dependent on the specific element and tabulated values like from NIST [112] exist. Above some Z (roughly argon, $Z = 18$) a rough approximation of $I = 10\,\text{eV} \cdot Z$ can be substituted [237], developed by Bloch [46].

The precise number for W strongly depends on the used gas mixture and typically requires experimental measurements to determine. Monte Carlo based simulations can be used as a rough guide, but care must be taken interpreting the results as significant uncertainty can remain. Tools for such Monte Carlo simulations include GEANT4 [5] [7] and MCNelectron [173] [8]. These are based on atomic excitation cross sections, which are well tabulated in projects like ENDF [54] (citation for the latest data release) [9] and LXCat [160, 171, 58] [10].

[7] https://geant4.cern.web.ch

[8] http://web.vu.lt/ff/a.poskus/mcnelectron/

[9] https://www.nndc.bnl.gov/endf-b8.0/index.html

[10] https://lxcat.net

5.3.3. Mean free path

While we have already discussed the mean free path for X-rays in sec. 5.1.1, we should revisit it quickly for electrons in a gas. It is a necessary concept when trying to understand other gaseous detector physics concepts, in particular for the gas amplification. [11]

The mean free path of an electron in a gas can be described by

$$\lambda = \frac{1}{n\sigma} \tag{5.7}$$

where $n = N/V$ is the number density (atoms or molecules per unit volume) of the gas particles and σ the cross section of electrons interacting with them. Such cross sections are tabulated for different elements, for example again see LXCat [160, 171, 58].

Based on the ideal gas law, we can express it as a function of pressure p and temperature T to

$$pV = Nk_BT \Leftrightarrow p = nk_BT,$$

with the Boltzmann constant k_B. Inserting this via n into eq. 5.7 yields

$$\lambda = \frac{k_BT}{p\sigma}. \tag{5.8}$$

5.3.4. Diffusion

Diffusion in the context of gaseous detectors is the process of the random walk of electrons either in longitudinal or transverse (to the electric field) direction they exhibit, when drifting towards the readout. Depending on the specific detector technology, some transverse diffusion may be a desired property. For long drift distances, magnetic fields can be used to significantly reduce the transverse diffusion.

For an overview of the mathematics of diffusion as a random walk process, see [39]. In the context of gaseous detectors see for example any of [186, 132, 106] In general, for precise numbers measurements must be taken. Monte Carlo simulations using tools like Magboltz [42] or PyBoltz [27] can be used as a general reference if no measurements are available.

The mean distance σ_t from a starting point after a certain amount of time t is given by

$$\sigma_t = \sqrt{2NDt}$$

for a random walk in N dimensions, where D is a diffusion coefficient specifying the movement in distance squared per time. For example for a point like source of multiple electrons, after diffusion

[11] Its usefulness depends on the angle from which different aspects are discussed of course. We mainly use it to understand the gas amplification later.

of some distance a 2 dimensional gaussian distribution will be expected with a standard deviation of σ_t. By relating the time to the drift velocity v along an axis and introducing the diffusion constant D_t

$$D_t = \sqrt{\frac{2ND}{v}}$$

we can further express σ_t as

$$\sigma_t = D_t \cdot \sqrt{x}, \tag{5.9}$$

which is often practically useful to estimate the expected diffusion after a certain drift length.

Note that the terminology of "diffusion constant", "diffusion coefficient" and similar is often used ambiguously as to whether they refer to D or D_t (or sometimes even σ_t). Nor is the considered dimensionality N always clearly indicated. Keeping $N = 1$ and handling multiple dimensions as independent random walks is a practical approach to take (as long as it is valid in the application). [12]

5.3.5. Drift velocity

The drift velocity is the average speed at which electrons move towards the anode in a gaseous detector under the influence of an electric field. It is required to understand how recorded time information relates to longitudinal shape information of recorded data. Based on the so called 'friction force model' an analytical expression for the drift velocity in an electromagnetic field can be written to:

$$\vec{v} = \frac{e}{m_e}\frac{\tau}{1+\omega^2\tau^2}\left(\vec{E} + \frac{\omega\tau}{B}(\vec{E}\times\vec{B}) + \frac{\omega^2\tau^2}{B^2}(\vec{E}\cdot\vec{B})\vec{B}\right)$$

with the electron charge e and mass m_e, in an electric field $\vec{E}$ and magnetic field $\vec{B}$, given the Lamor frequency $\omega = eB/m_e$ and the mean collision time τ. [237]

For detectors without a magnetic field $B = 0, \omega = 0$ and a constant, homogeneous electric field E, this reduces to the Townsend expression:

$$v = \frac{eE\tau}{m_e}.$$

If measurements are not available, these can also be computed by Magboltz [42] or PyBoltz [27], which solve the underlying transport equation, the Boltzmann equation.

[12] Later in chapter 11.4.2 we will discuss Monte Carlo event generation, which use the diffusion coefficient as an input to generate clusters after drift. Distance x and y are each sampled individually according to eq. 5.9 and combined to a radial distance from the cluster center.

5.3.6. Gas amplification

In order to turn the individual electrons into a measurable signal, gaseous detectors use some form of gas amplification. Details vary, but it is usually a region in the gas volume close to the readout with a very strong electric field (multiple kV cm^{-1}) such that each electron causes many secondary ionizations, leading to an avalanche effect. In case of the detectors described in this thesis, amplification factors between 2000 to 4500 are desired.

An electron in a strong electric field $\vec{E}$ will gain enough energy to ionize an atom of the gas after a distance

$$l \geq \frac{I}{|\vec{E}|}$$

with the ionization potential of the gas, I. This needs to be put into relation with the mean free path, eq. 5.8. If $l \ll \lambda$ no secondary ionization takes place and if $l \gg \lambda$ every interaction leads to ionization, likely resulting in a breakdown causing an arc (see also Paschen's law [161], not covered here). In the intermediate range some interactions cause secondary ionization, some does not.

We can make the statistical argument that

$$\mathcal{N} = e^{-l/\lambda}$$

is the relative number of collisions with $l > \lambda$. This allows to define the probability of finding a number of ionizations per unit length to be

$$P(l) = \frac{1}{\lambda} e^{-l/\lambda} = \alpha,$$

where we introduce the definition of the 'first Townsend coefficient', α. We can insert the definition of the mean free path, eq. 5.8, and l into this equation to obtain

$$\alpha(T) = \frac{p\sigma}{kT} \exp\left(-\frac{I}{|\vec{E}|} \frac{p\sigma}{kT} \right). \tag{5.10}$$

With this we have an expression for the temperature and pressure dependency of the first Townsend coefficient. [13] This derivation followed [187] based on [82], but see [18] for a more general treatment. Similarly to diffusion and drift parameters, the first Townsend coefficient can be computed numerically using tools like Magboltz.

Note that [186] introduces the Townsend coefficient as $\alpha = \frac{1}{\lambda}$, introducing a specific λ and σ referring to the *ionization* mean free path and *ionizing* cross sections. This can be misleading as it makes it seem as if the Townsend coefficient is inversely proportional to the temperature. This is of course only the case in the regime where each interaction actually causes another ionization.

[13] This will be a useful reference later when discussing possible temperature effects of the detector operated at CAST.

The first Townsend coefficient can be used to express the multiplication factor – or gas amplification – used in a detector,

$$G = e^{\alpha x}$$

as the number increases by α after each step dx.

The statistical distribution describing the number of electrons after gas amplification is the Pólya distribution

$$p(x) = \frac{N}{G}\frac{(1+\theta)^{1+\theta}}{\Gamma(1+\theta)}\left(\frac{x}{G}\right)^{\theta}\exp\left[-\frac{(1+\theta)x}{G}\right] \tag{5.11}$$

where N is a normalization constant, θ is another parameter performing shaping of the distribution and G is the effective gas gain. Γ refers to the gamma function. It is to note that the term ”polya distribution” in this context is different from other mathematical definitions, in which polya distribution usually means a negative binomial distribution. The above definition goes back to Alkhazov [6] and in slight variations is commonly used. Due to the complexity of this definition, care needs to be taken when performing numerical fits to data with this function (using bounded parameters and preferring more general non-linear optimization routines instead of a normal Levenberg-Marquardt non-linear least squares approach).

Based on eq. 5.10 the largest impacts on the expected gas amplification have the electric field, the choice of gas and the temperature of the gas. While the former two parameters are comparatively easy to control, the temperature in the amplification region may vary and is hard to measure. As such depending on the detector details and application, gas gain variations are expected and corrections based on a running gas gain value may be necessary.

As the large number of interactions in the amplification region can excite many atoms of the (typically) noble gas, UV photons can be produced. Their mean free path is comparatively long relative to the size of the amplification region. They can start further avalanches, potentially away from the location of the initial avalanche start, lowering spatial resolution and increasing apparent primary electron counts. Molecular gases are added – often only in small fractions – to the gas mixture to provide rotational or vibrational modes to dissipate energy without emitting UV photons. In this context the molecular additions are called ’quencher gases’.

5.3.7. Energy resolution

Because of the statistical processes associated with the full detection method used in gaseous detectors, even a perfect delta like signal in energy, will lead to a natural spread of the measured signal. The convolution of different ionization yields, potential losses and gas gain variation all contribute to such a spread.

As such a typical measure of interest for gaseous detectors is the energy resolution, which is commonly defined by

$$\Delta E = \frac{\sigma}{E}$$

where σ is the standard deviation of the normal distribution associated with the resolved peak in the detectors data, assuming a – for practical purposes – delta peak as the input spectrum. Sometimes definitions using the full width at half maximum (FWHM) are also used in place of σ. Typical values for the energy resolution defined like this are smaller than 15 %.

If the absolute magnitude of the σ at a given energy is constant, which at least is partially reasonable as the width is not fully due to energy dependent effects, the energy resolution is proportional to $1/E$.

The Fano factor [84], defined by the variance over the mean of a distribution $F = \sigma^2/\mu$ (typically within some time window), improves the ideal energy resolution. It arises in the associated statistical processes, because there are a finite number of possible interaction states. For X-rays an additional aspect is due to the maximum energy transferable into the gas due to the X-rays energy. In practice though the energy resolution of gaseous detectors is usually limited by other effects.

5.3.8. Escape photons and peaks | 55Fe as a typical calibration source

Finally, gaseous detectors need to be calibrated, monitored and tested. This is commonly done with a ^{55}Fe source. ^{55}Fe is a radioactive isotope of iron, which decays under inverse beta decay to ^{55}Mn. Due to the required restructuring of the electronic shells, the manganese is in an excited state. While the emission of an Auger electron with 5.19 keV dominates with a probability of 60 %, as an X-ray source the Kα_1 and Kα_2 lines with combined energies of about 5.9 keV are of note.

When using such a ^{55}Fe source as a calibration source for an argon filled gaseous detector, the 5.9 keV photons will produce a photoelectron in argon. Mostly an inner shell electron will be liberated, producing a photoelectron of around 2.7 keV. The excited argon atom can now emit an Auger electron in about 85 % of cases, which will fully deposit its remaining energy into the surrounding gas via further ionizations, resulting in the 'photopeak' at around 5.9 keV.

If however another photon is produced with an energy below the $K1s$ energy of argon (3.2 keV) – for example via Kα_1 or Kα_2 fluorescence of argon, both at about 2.95 keV – such a photon has a very long absorption length in the gas volume, about $l_{\text{abs}} = 3.5\,\text{cm}$ (cf. fig. 5.1). This can cause it to easily escape the active detection region, especially if the sensitive region of the detector is comparatively small. The result is a measured signal of $E_i - E_k = 5.9\,\text{keV} - 2.95\,\text{keV} \approx 2.9\,\text{keV}$, called the 'escape peak'. [186, 132, 112]

Such an additional escape peak is useful as a calibration tool, as it gives two distinct peaks in a

^{55}Fe calibration spectrum, which can be utilized for an energy calibration. One important consideration of such an escape peak is however, that this escape peak of about 3 keV is not equivalent to a real 3 keV X-ray. As the original X-ray is still the 5.9 keV photon with its distinct absorption length, the geometric properties of ensembles of escape peak events and real 3 keV photons is different. This will be important later in sec. 11.4.7.

6. Septemboard detector Detector

Contents

With the theoretical aspects of gaseous detectors out of the way, in sec. 6.1 we will introduce the 'Micromegas' type of gaseous detector. Micromegas can be read out in different ways. One option is the Timepix ASIC, sec. 6.2, by way of the 'GridPix', sec. 6.3.

The GridPix detector in use in the 2014 / 2015 CAST data taking campaign, to be presented in section 6.4, had a few significant drawbacks for more sensitive searches. In particular for searches at low energies $\lesssim 2\,\text{keV}$ and searches requiring low backgrounds over larger areas on the chip (for example the chameleon search done in [134]). For this reason a new detector was built in an attempt to improve each shortcoming of the detector.

We introduce the 'Septemboard' detector with a basic overview in section 6.5. From there we continue looking at each of the new detector features motivating their addition. All detector upgrades were done to alleviate one or more of the old detector's drawbacks. For each new detector feature we will highlight the aspects it is intended to improve on.

Section 6.7 introduces two new scintillators as vetoes. These require the addition of an external shutter for the Timepix, which is realized by usage of a flash ADC (FADC), see section 6.8. Further, an independent but extremely important addition is the replacement of the Mylar window by a silicon nitride window, section 6.9. Another aspect is the addition of 6 GridPixes around the central GridPix, the 'Septemboard' introduced in section 6.10. Due to the additional heat produced by 6 more GridPix, a water cooling system is used, sec. 6.11. Lastly, in sec. 6.12 we will also discuss the combined detection efficiency of this detector.

6.1. Micromegas working principle

Micro Mesh **Ga**seous **S**tructures (Micromegas) are a kind of **M**icro**p**attern **G**aseous **D**etectors (MPGDs) first introduced in 1996 [92, 91]. The modern Micromegas is the Microbulk Micromegas [12].

Interestingly, the name Micromegas is based on the novella Micromégas by Voltaire published in 1752 [219], an early example of a science fiction story. [92]

These detectors are – as the name implies – gaseous detectors containing a 'micro mesh'. In the most basic form they are a made of a closed detector volume that is filled with a suitable gas, allowing for ionization (often argon based gas mixtures are used; xenon based detectors for axion helioscopes are in the prototype phase). The volume is split into two different sections, a large drift volume typically $\mathcal{O}$(few cm) and an amplification region, sized $\mathcal{O}$(50 to 100 μm). At the top of the volume is a cathode to apply an electric field. Below the mesh is the readout area at the bottom of the volume. In standard Micromegas detectors, strips or pads are used as a readout. The electric field in the drift region is strong enough to avoid recombination of the created electron-ion pairs and to provide reasonably fast drift velocities $\mathcal{O}(\mathrm{cm}\,\mu\mathrm{s}^{-1})$. The amplification gap on the other hand is precisely used to multiply the primary electrons using an avalanche effect. Thus, the electric field reaches values of $\mathcal{O}(50\,\mathrm{kV\,cm}^{-1})$. These drift and amplification volumes are achieved by an electric field between a cathode and the mesh as well as the mesh and the readout area.

Fig. 6.1 shows a schematic for the general working principle of such detectors

6.2. Timepix ASIC

The Timepix ASIC (Application Specific Integrated Circuit) is a 256×256 pixel ASIC with each pixel $55 \cdot 55\,\mu\mathrm{m}^2$ in size. It is based on the Medipix ASIC, developed for medical imaging applications by the Medipix Collaboration [141]. The pixels are distributed over an active area of $1.41 \times 1.41\,\mathrm{cm}^2$. Each pixel contains a charge sensitive amplifier, a single threshold discriminator and a 14 bit pseudo random counting logic. It requires use of an external clock, in the range of 10 to 150 MHz, with 40 MHz being the clock frequency used for the applications in this thesis. [145, 142, 143] A good

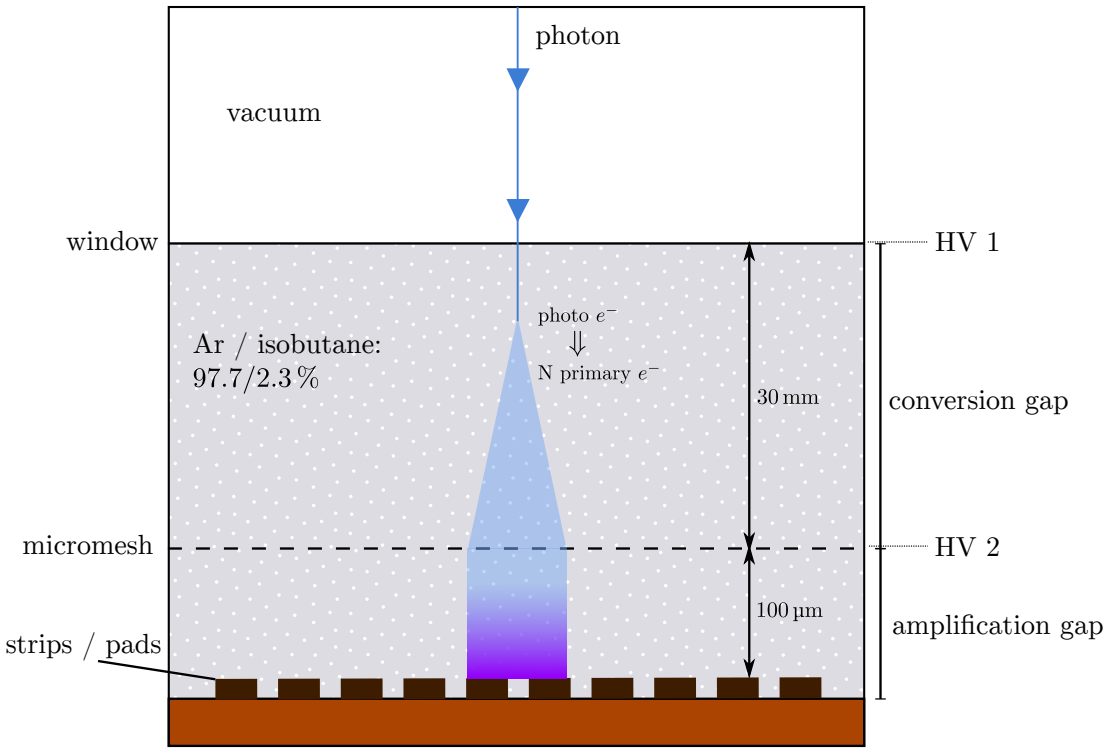


Figure 6.1.: Working principle of a general Micromegas detector. Specific distances and gas mixture are exemplary. An ionizing photon enters through the detector window into the gas-filled detector body. After a certain distance it produces a photoelectron, which ionizes further gas molecules for a specific number of primary electrons (depending on the incoming photon's energy) and gas mixture. The primary electrons drift towards the micromesh due to the drift voltage, thereby experiencing diffusion. In the much higher voltage in the amplification gap an avalanche of electrons is produced, enough to trigger the readout electronics (strips or pads).

overview of the Timepix is also given in [146]. A picture of a Timepix ASIC is shown in fig. 6.2(a).

The Timepix uses a shutter based readout, either with a fixed shutter time or using an external trigger to close a frame. After the shutter is closed in the Timepix, the readout is performed during which the detector is insensitive. Each pixel further can work in four different modes:

hit counting mode / single hit mode simply counts the number of times the threshold of a pixel was crossed (or whether a pixel was activated once in single hit mode).

Time over Threshold (ToT) In the ToT mode the counter of a pixel will count the number of clock cycles that the charge on the pixel exceeds the set threshold, which is set by an 8 bit **D**igital to **A**nalog **C**onverter (DAC) while the shutter is open. ToT is equivalent to the collected charge of each pixel.

Time of Arrival (ToA) The ToA mode records the number of clock cycles from the first time the pixel's threshold is exceeded to the end of the shutter window. Thus, it allows to calculate the time at which the pixel first crossed the threshold.

In the context of this thesis only the ToT mode was used.

6.2.1. Timepix3

The Timepix3 is the successor of the Timepix. [172, 144] It is generally similar to the Timepix, but would provide 3 important advantages if used in a gaseous detector for the applications in this thesis:

- clock rates of up to 300 MHz for higher possible data rates (less interesting for data taking in an axion helioscope)
- a stream based data readout. This means no fixed shutter times and no dead time during readout. Instead data is sent out when it is recorded in parallel.
- each pixel can record ToT *and* ToA information at the same time. This allows to record the charge recorded by a pixel as well as the time it was activated, yielding 3D event reconstruction with precise charge information.

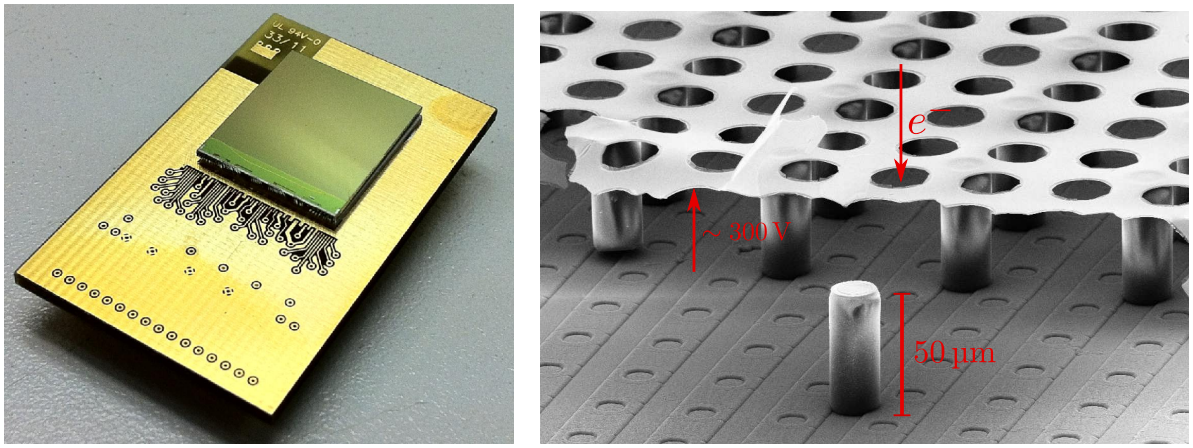
An open source readout was developed by the University of Bonn and is available under [97] [1]. A gaseous detector based on this is currently in the prototyping phase, see the upcoming [188].

[1] https://github.com/SiLab-Bonn/tpx3-daq

6.3. GridPix

First experiments of combining a Micromegas with a Timepix readout were done in 2005 [56] by using classical approaches to place a micromesh on top of the Timepix, at the time still called *TimePix-Grid*. While this worked in principle, it showed a Moiré pattern, due to slight misalignment between the holes of the micromesh and the pixels of the Timepix. Shortly after, an approach based on photolithographic post-processing was developed to perfectly align the Timepix pixels each with a hole of a micromesh [59], called the *InGrid* (integrated grid). The commonly used name for a gaseous detector using an InGrid is GridPix. For an overview of the process to produce InGrids nowadays, see [187].

The InGrid consists of a 1 µm thick aluminum grid, resting on small pillars 50 µm above the Timepix. A silicon-nitride $Si_x N_y$ [2] layer protects the Timepix from direct exposure to the amplification processes. The main advantage over previous Micromegas technologies of the GridPix is its ability to detect single electrons. As long as the diffusion distance is long enough to avoid multiple electrons entering a single hole of the InGrid, each primary electron produced during the initial ionization event is recorded. Fig. 6.2(b) shows an image of such an InGrid.



(a) Timepix ASIC (b) InGrid

Figure 6.2.: (a) Picture of a bare Timepix ASIC. (b) Image of an InGrid, which was partially cut for inspection under an electron microscope. The pillars support the micromesh and have a height of 50 µm. Each hole is perfectly aligned with a pixel of the Timepix below. Typical voltages applied between the grid and the Timepix are shown.

[2] The x, y notation is sometimes encountered when the exact material composition is not known. Silicon-nitride is a term used for a range of silicon to nitride ratios. Most commonly $Si_3 N_4$.

6.4. 2014 / 2015 GridPix detector for CAST

In the course of [134] a first GridPix based detector for usage at an axion helioscope, CAST, was developed. While the main result was on the coupling constant of the chameleon particle, an axion-electron coupling result was computed in [192].

The detector consists of a single GridPix in a 78 mm diameter gas volume and a drift distance of 3 cm. The detector has a 2 µm thick Mylar ($C_{10} H_8 O_4$) entrance window for X-rays. This detector serves as the foundation the detector used in the course of this thesis was built on. See fig. 6.3 for an exploded schematic of the detector. Further, fig. 6.4 shows the achieved background rate of this detector in the center $5 \times 5\,\mathrm{mm}^2$ region of the detector. The background rate shows the copper Kα line near 8 keV, possibly overlaid with a muon contribution as well as the expected argon Kα lines at 3 keV. Below 2 keV the background rises the lower the energy becomes, likely due to background- and signal-like events being less geometrically different at low energies (fewer pixels). The average background rate in the range from 0 to 8 keV is $\sim 2.9 \times 10^{-5}\,\mathrm{keV}^{-1}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$.

6.5. Septemboard detector overview

Generally, the detector follows the same design as the old detector shown in sec. 6.4, mainly so that mounting it inside of the lead shielding and to the vacuum pipes at CAST is possible without significant changes. An exploded view of the full detector can be seen in fig. 6.5.

At the center of the new detector is the 'septemboard', 7 GridPixes replace the single GridPix on the carrier board, sec. 6.10. Analogue signals induced by the amplified charges on the center GridPix are now read out using a flash ADC (FADC), sec. 6.8. The housing with an inner diameter of 78 mm is again made of acrylic glass, same as in the old detector. The detector entrance window is replaced by a 300 nm $Si_3 N_4$ window (sec. 6.9), which also acts as part of the detector cathode. The copper anode slots in right above the septemboard. The carrier board sits on the intermediate board. Below the intermediate board is a bespoke water cooling made of oxygen-free copper to cool the heat emitted by the additional 6 GridPixes, sec. 6.11. On the underside of the intermediate board is a new small silicone photomultiplier (SiPM), sec. 6.7. Finally, a large veto scintillator is installed above the detector setup at CAST, also sec. 6.7.

During developments multiple septemboards were built and tested. The septemboard used in the final detector is septemboard 'H'. The GridPixes of the final board are listed in tab. 6.1.

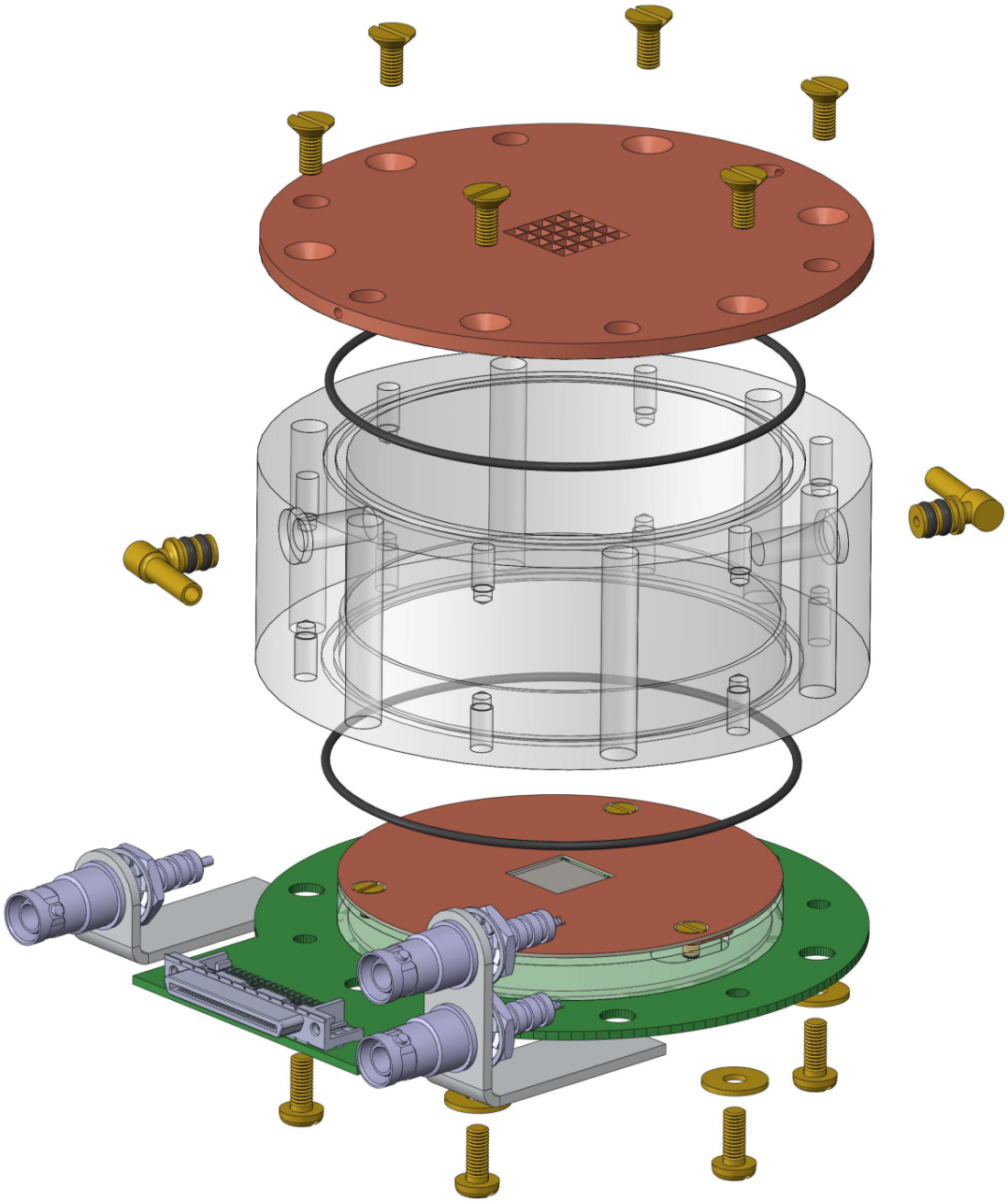


Figure 6.3.: Exploded view of the GridPix detector used during the 2014/15 data taking campaign at CAST. Consists of a 3 cm drift volume with a 78 mm inner diameter and a single GridPix at the center.

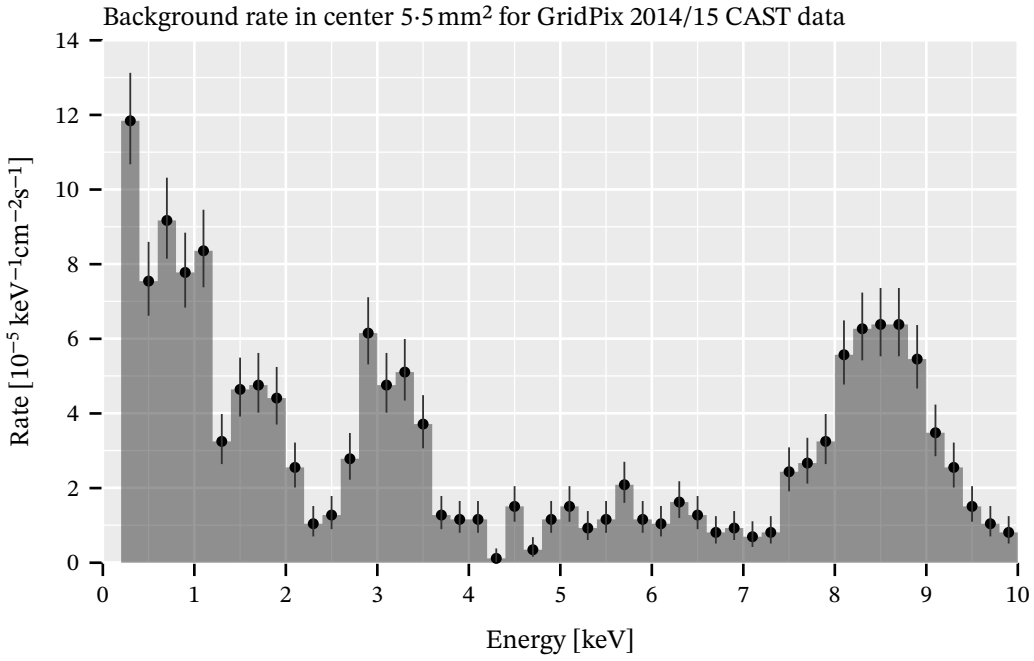


Figure 6.4.: Background rate in the center $5 \times 5\,\mathrm{mm}^2$ for the GridPix used in 2014/15 at CAST. It corresponds to a background rate of $\sim 2.9 \times 10^{-5}\,\mathrm{keV}^{-1}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$ in the range from 0 to 8 keV.

Chip	Number
E 6 W69	0
K 6 W69	1
H 9 W69	2
H10 W69	3
G10 W69	4
D 9 W69	5
L 8 W69	6

Table 6.1.: Overview of the different chips on septemboard H. The first part of the name corresponds to the position on the wafer and `W69` is Timepix wafer number 69.

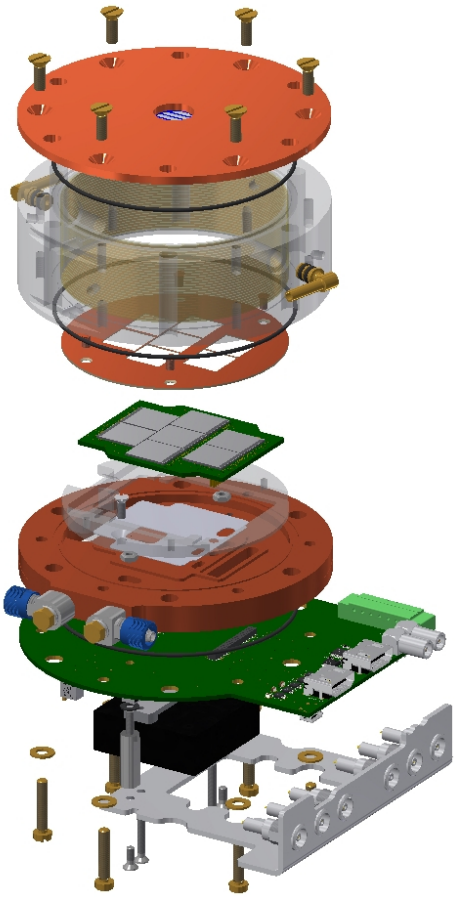


Figure 6.5.: Exploded view of the main GridPix septemboard detector. The FADC and large veto scintillator paddle are not shown for obvious reasons. At the center of the detector is the 'septemboard', 7 GridPixes on a carrier board. The housing is made of acrylic glass, same as in the old detector. The top shows the 300 nm $Si_3 N_4$ window. Below the intermediate board is the water cooling made of pure copper. At the bottom, the SiPM veto scintillator can be seen.

6.6. Detector readout system

The detector is operated by a Xilinx Virtex-6 **F**ield **P**rogrammable **G**ate **A**rray (FPGA) in the form of a Virtex-6 ML605 evaluation board. [3] It is connected to the intermediate board via two **H**igh-**D**efinition **M**ultimedia **I**nterface (HDMI) cables. The Virtex-6 contains the firmware controlling the Timepix ASICs and correlating the scintillator and FADC signals (see appendix sec. A.1), the Timepix Operating Firmware (TOF). The high voltage (HV) supply both for the septemboard as well as for the scintillators sit inside a VME crate, which also houses the FADC. A USB connection is used to read out and control the FADC and HV supply via the computer running the data acquisition and control software (see sec. A.2), the Timepix Operating Software (TOS). A schematic of this setup is shown in fig. 6.6, which leaves out the SiPM and temperature readout.

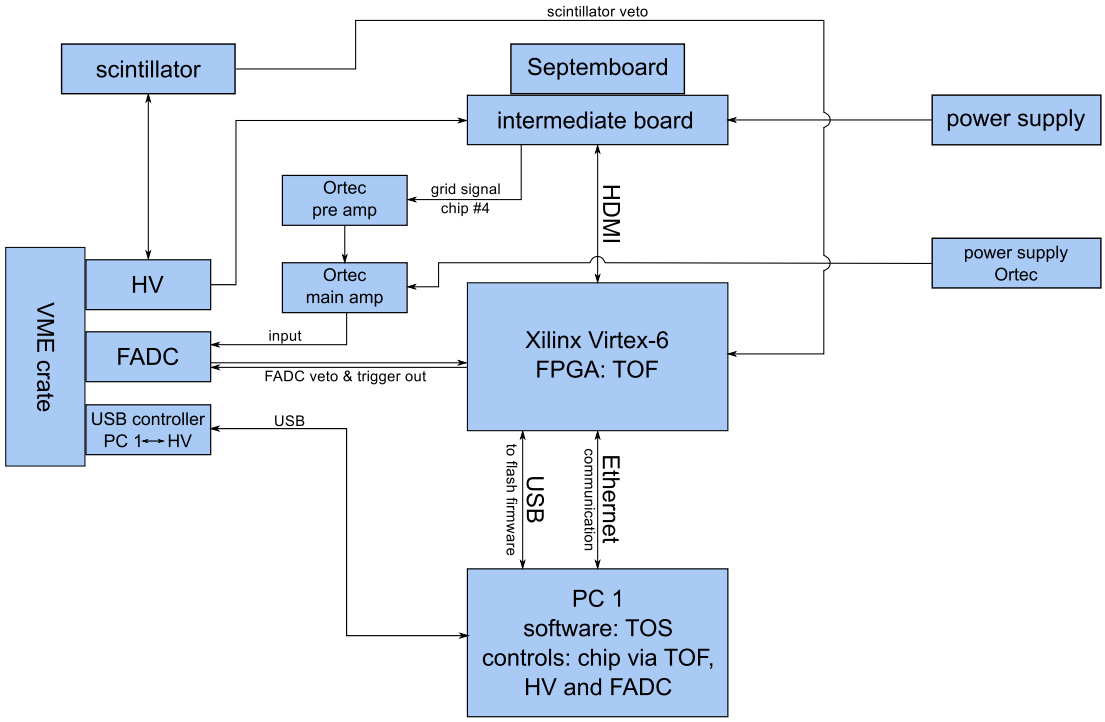


Figure 6.6.: Flowchart of the whole detector and readout system

6.7. Scintillator vetoes

The first general improvement is the addition of two scintillators for veto purposes. While both have slightly different goals, each is there to help with the removal of muon signals or muon induced events (for example X-ray fluorescence) in the detector. Given that cosmic muons (ref. section 5.2) dominate the background by flux, statistically there is a high chance of muons creating X-ray like

[3] `https://www.xilinx.com/products/boards-and-kits/ek-v6-ml605-g.html` (visited 2022/10/17)

signatures in the detector. By tagging muons before they interact near the detector, these can be correlated with events seen on the GridPix and thus possibly be vetoed if precise time information is available.

The first scintillator is a large 'paddle' installed above the detector and beamline, aiming to tag a large fraction of cosmic muons traversing in the area around the detector. It has a Canberra 2007 base and the photomultiplier tube (PMT) is a Bicron Corp. 31.49x15.74M2BC408/2-X (first two numbers: dimensions in inches). The full outer dimensions of the scintillator paddle are closer to 42 cm×82 cm. It is the same scintillator paddle used during the Micromegas data taking behind the LLNL telescope prior and after the data taking campaign with the detector described in this thesis.

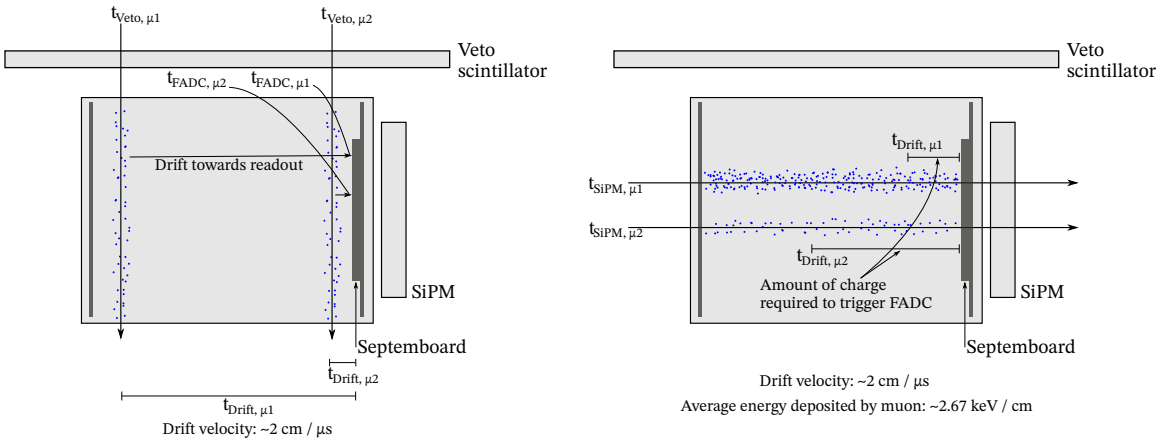
For this scintillator, muons which traverse through it and the gaseous detector volume are not the main use case. They can be easily identified by the geometric properties of the induced tracks (their zenith angles are relatively small, resulting in track like signatures as the GridPix readout is orthogonal to the zenith angle). There is a small chance however that a muon can ionize an atom of the detector material, which may emit an X-ray upon recombination. One particular source of background can be attributed to the presence of copper whose Kα lines are at $\sim$ 8.04 keV as well as fluorescence of the argon gas with its Kα lines at $\sim$ 2.95 keV (see. table 5.1 in sec. 5.1.4 and sec. 5.1.4).

Fig. 6.7(a) shows a schematic of a side view of the detector chamber with the scintillator paddle on top. When a muon traverses the scintillator, a counter t_{Veto} starts on the FPGA. Two different cases are shown. In the extreme, a muon may traverse close to the cathode or close to the anode / readout plane. This changes the time of the total drift time and therefore the time difference between the trigger time t_{Veto} and the readout time, which in the readout is precisely the value of the counter on the FPGA. The drift velocity of $\sim$ 2 cm μs^{-1} and height of the detector chamber (3 cm) therefore allow to set an upper limit on the maximum time between a veto paddle trigger and the GridPix readout of about 1.5 μs. At a clock speed of 40 MHz this corresponds to 60 clock cycles, a number we will later try to see in the data (see sec. 11.5.1]]). As the location at which the muon traverses through the detector is random and homogeneous throughout the detection volume, we expect to see a flat distribution up to the maximum possible time and then a sharp drop (equivalent to muons at the cathode).

The second scintillator is a small silicon photomultiplier (SiPM) installed on the underside of the PCB on which the septemboard is installed. This scintillator was calibrated and set up as part of [196]. We are interested in tagging precisely those muons, which enter the detector orthogonally to the readout plane. This implies zenith angles of almost 90 ° such that the elongation in the transverse direction of the muon track is small enough to result in a small eccentricity. From the Bethe equation the mean energy loss of muons in the used gas mixture is about 8 keV along the 3 cm of drift volume in the detector (see fig. 5.2 for the energy loss). This coincides with the copper Kα lines and should

lead to another source of background in this energy range. Although the muon background will have a much wider energy distribution than the copper lines, which are dominated by the energy resolution of the detector. In a similar manner to the veto paddle we can make an estimate on the typical time scales associated from the time of the scintillator trigger to the GridPix detection. A muon that traverses orthogonally trough the detector can be taken to leave an instant ionization track and trigger the SiPM at the same time ($\mathcal{O}(100\,\text{ps}) \ll 25\,\text{ns}$ for one clock cycle). As such, the relevant time scale is the drift time until enough charge has drifted towards the grid as to pass the activation threshold.

Ionization of a muon is a statistical process, as indicated in fig. 6.7(b). Depending on the density of the charge cloud for muons orthogonal to readout plane, time to accumulate enough charge to trigger FADC differs. With an average energy deposition of a muon in argon gas of $\sim 2.67\,\text{keV}\,\text{cm}^{-1}$, and drift velocity again of $2\,\text{cm}\,\mu\text{s}^{-1}$ the accumulation time can be estimated. For example assuming an FADC activation threshold of 1.5 keV the necessary charge is accumulated on 0.56 cm, which takes about 280 ns $\approx$ 11 clock cycles to accumulate. Different tracks will have deposited different amounts of energy. Therefore, we expect a peak at relatively low clock cycles with a tail up to the same 60 clock cycles (in case the full 3 cm track needs to be accumulated to activate the FADC).



(a) Veto paddle

(b) SiPM

Figure 6.7.: (a)Schematic of expected signals for different muons from the zenith passing through scintillator paddle and detector. t_{Veto} marks beginning of a counter. Where the muon traverses changes drift time and thus time difference between two times. (b)Ionization of a muon is a statistical process. Depending on the density of the charge cloud for muons orthogonal to readout plane, time to accumulate enough charge to trigger FADC differs.

6.8. FADC

As the Timepix is read out in a shutter based fashion and typical shutter lengths for low rate experiments are long compared to the rate of cosmic muons, the scintillators introduced in previous section require an external trigger to close the Timepix shutter early if a signal is measured on the Timepix. This is one the main purposes of the flash **a**nalog to **d**igital **c**onverter (FADC) that is part of the detector. This is done by decoupling the induced analogue signals from the grid of the center GridPix.

The specific FADC used for the detector is a Caen V1792a. It runs at an internal 50 MHz or 100 MHz clock and utilizes virtual frequency multiplication to achieve sampling rates of 1 GHz or 2 GHz, respectively. It has 4 channels, each with a cyclic register of 2560 channels. At an operating clock frequency of 1 GHz that means each channel covers the last $\sim 2.5\,\mu s$ at any time. [55]

The raw signal decoupled from the grid is first fed into an Ortec 142 B pre-amplifier and then feeds into an Ortec 474 shaping amplifier, which integrates and shapes the signal as well as amplifies it. For a detailed introduction to this FADC system, see the thesis of A. Deisting [75] and [192] for further work integrating it into this detector. In addition see the FADC manual [55] [4] for a deep explanation of the working principle of this FADC.

The analogue signal of the center grid is decoupled via a small $C_{\text{dec}} = 10\,\text{nF}$ capacitor in parallel to the high voltage line. For a schematic of the circuit see fig. 6.8. When a primary electron traverses through a hole in the grid and is amplified, the back flowing ions induce a small voltage spike on top of the constant high voltage applied to the grid. The parallel capacitor filters out the constant high voltage and only transmits the time varying induced signals. Such signals – the envelope of possibly many primary electrons – are measured by the FADC.

This signal can be used for three distinct things:

1. it may be used as a trigger to close the shutter of the ongoing event. Ideally, we want to only measure a single physical event within one shutter window. A long shutter time can statistically result in multiple events happening, which the FADC trigger helps to alleviate. This allows us to reduce the number of events with multiple physical events and acts as a trigger for the scintillators. This in turn means possible muon induced X-ray fluorescence can be vetoed.

2. By nature of the signal production and drift properties of the primary electrons before they reach the grid, the signal shape can theoretically be used to determine a rough longitudinal shape of the event. The length of the FADC event should be proportional to the size of the primary electron cloud distribution along the 'vertical' detector axis. This potentially allows to differentiate between a muon traversing orthogonally through the readout plane and an X-ray

[4] https://archive.org/details/manualzilla-id-5646050/

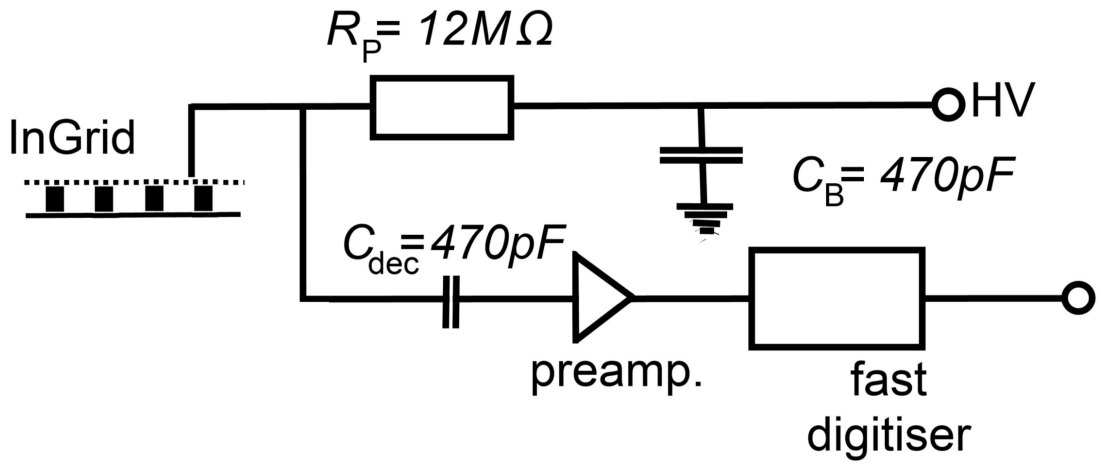


Figure 6.8.: Schematic of the setup to decouple signals induced on the grid of the InGrid. The signal is decoupled in the sense that the capacitor essentially acts as a low pass filter, thus removing the constant HV. Only the high frequency components of the induced signals on top of the HV pass into the branch leading to the FADC. In the detector of this thesis, a capacitance of 10 nF was used instead. The decoupling is implemented on the intermediate board. Schematic taken from [75].

due to their longitudinal shape difference, see sec. 11.5.2.

3. Finally, it provides an independent measure of the collected charge on the center chip. This will prove useful in understanding the detector behavior over time later in sec. 10.2.1.

The working principle of how the FADC and the scintillators can be used together to remove certain types of background, by correlating events in the scintillators, the FADC and the GridPix, is shown in fig. 6.9.

6.9. SiN window

Next up, a major limitation of the previous detector was its limited combined efficiency below 2 keV, due to its 2 μm Mylar window. Therefore, the next improvement for the new detector is an ultra-thin silicon nitride Si_3N_4 window of 300 nm thickness and 14 mm diameter, developed by Norcada [5]. A strongback support structure consisting of 4 lines of 200 μm thick and 500 μm wide Si_3N_4, helps to support a pressure difference of up to 1.5 bar. On the outer side a 20 nm thin layer of aluminum is coated to allow the window to be part of the detector cathode. The strongback occludes about 17 % of the full window area. In reality it is slightly more, as the strongbacks become somewhat wider towards the edges. In the centermost region they are straight and in the center $5 \times 5\,\text{mm}^2$ area, they occlude 22.2 %.

[5] https://www.norcada.com/

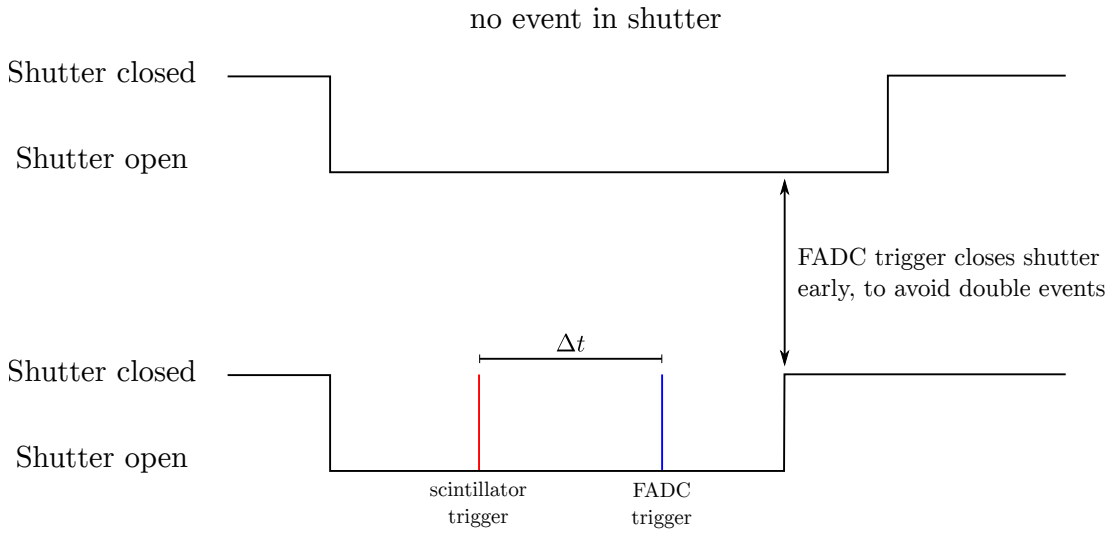


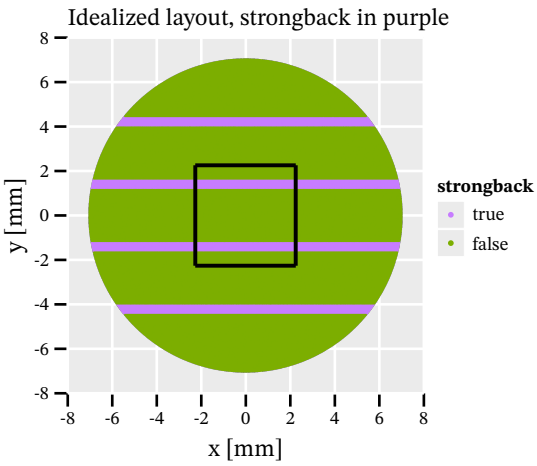
Figure 6.9.: Schematic showing how the FADC and scintillators are used together to tag possible coincidence events and close the shutter early to reduce the likelihood of multi-hit events. If the scintillator triggers when the shutter is open, a clock starts counting up to 4096 clock cycles. On every new trigger this clock is reset. If the FADC triggers, the scintillator clock values are read out and can be used to correlate events in the scintillator with FADC and GridPix information. Further, the FADC trigger is used to close the Timepix shutter 5 µs after the trigger.

Fig. 6.10(a) shows the idealized strongback structure without a widening towards the edges of the window. Fig. 6.10(b) shows an image of one such window under testing conditions in the laboratory, as it withstands a pressure difference of 1.5 bar.

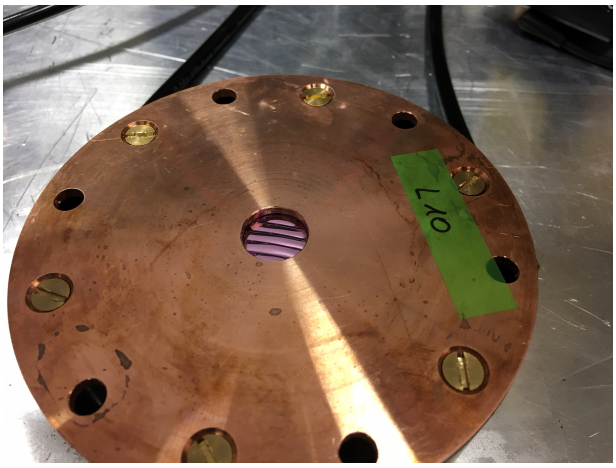
As the main purpose is the increase of transmission at low energies, fig. 6.11 shows the transmission of the mylar window of the old detector and the new $Si_3 N_4$ window in the energy range below 3 keV. The $Si_3 N_4$ window shows a significant increase in transmission below 2 keV, which is very important for the sensitivity in solar axion-electron and chameleon searches, which both peak near 1 keV in their solar flux. The window alone significantly increases the signal to noise ratio of these physics searches.

6.10. Septemboard - 6 GridPixes around a center one

The main motivation for extending the readout area from a single chip to a 7 chip readout is to reduce background towards the outer sides of the chip, in particular in the corners. Against common intuition however, it also plays a role for events which have cluster centers near the center of the readout. The latter is, because diffusion can produce quite large clusters even at low energies. In particular in lower energy events, tracks may have gaps in them large enough to avoid being detected as a single



(a) Window strongback schematic



(b) Image

Figure 6.10.: (a)shows an idealized schematic of the window strongback based on a simple MC simulation. 22.2 % of the area inside the inner $5 \times 5\,\mathrm{mm}^2$ area (black square) are occluded.(b)shows an image of one such window while testing in the laboratory, if it holds 1.5 bar. Image courtesy of Christoph Krieger.

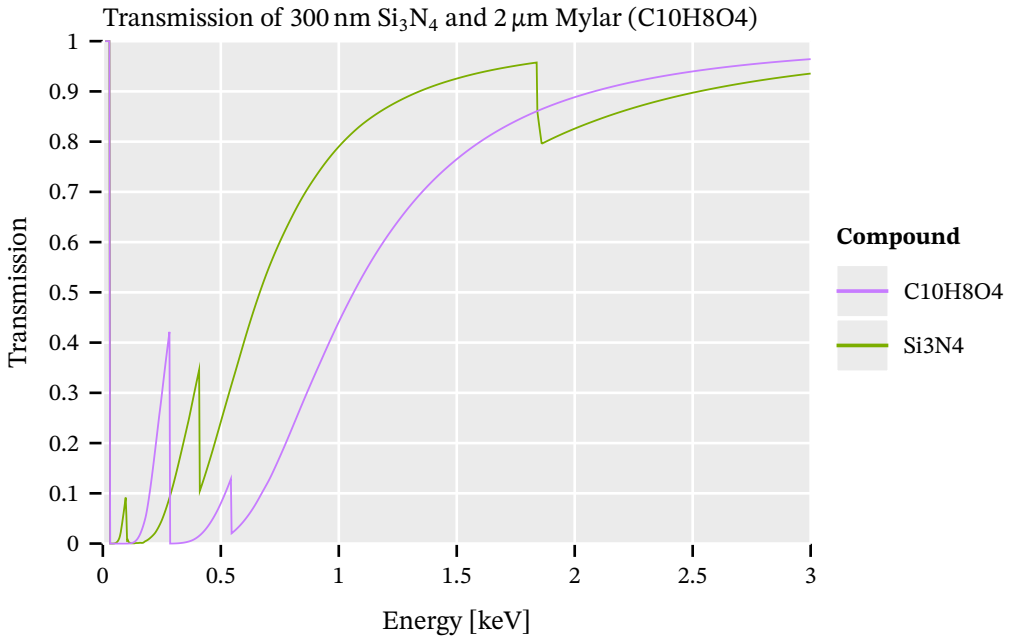


Figure 6.11.: Comparison of the transmission of a 2 µm Mylar window and a 300 nm Si_3N_4 window. The efficiency gains become more and more pronounced the lower the energy is, aside from the absorption edge of carbon at around 250 eV and above about 1.75 keV. In the interesting range around 1 keV significant transmission gains are achieved.

cluster for standard radii in cluster searching algorithms. This is particularly of interest as different searches produce an 'image' at different positions and sizes on the detector. While the center chip is large enough to fully cover the image for essentially all models, it may not be in the regions of lowest background. Hence, improvements to larger areas are needed.

The septemboard is implemented in such a way to optimize the loss of active area due to bonding requirements and general manufacturing realities. As the Timepix ASIC is a 16.1 mm by 14.1 mm large chip (the bonding area adding 2 mm on one side), the upper two rows are installed such that they are inverted to another. The bonding area is above the upper row and below the center row. The bottom row again has its bonding area below. This way the top two rows are as close together as realistically possible, with a decent gap on the order of 2 mm between the middle and bottom row. Any gap is potentially problematic as it implies loss of signal in that area, complicating the possible reconstruction methods. The layout can be seen in fig. 6.14 in the next section.

All 7 GridPix are connected in a daisy-chained way. This means that in particular for data readout, all chips are read out in serial order. The dead time for readouts therefore is approximately 7 times the readout time of a single Timepix. A single Timepix has a readout time of $\sim$ 25 ms at a clock frequency of 40 MHz (the frequency used for this detector). This leads to an expected readout time of the full septemboard of 175 ms. [6] Such a long readout time leads to a strong restriction of the possible applications for such a detector. Fortunately, for the use cases in a very low rate experiment such as CAST, long shutter times are possible, mitigating the effect on the fractional dead time to a large extent.

Fig. 6.12 shows a heatmap of all cluster centers during roughly 2000 h of background data after passing these clusters through a likelihood based cut method aiming to filter out non X-ray like clusters (details of this follow later in sec. 11.1). It is clearly visible that the further a cluster center is towards the chip edges, and especially the corners, the more likely it is to be considered an X-ray like cluster. This has an easy geometric explanation. Consider a perfect track traversing over the whole chip. In this case it is very eccentric. Move the same track such that its center is in one of the corners and rotate it by 45 ° and suddenly the majority of the track won't be detected on the chip anymore. Instead something roughly circular remains visible, 'fooling' the likelihood method. For a schematic illustrating this, see fig. 6.13.

The septemboard therefore is expected to significantly reduce the background over the whole center chip, with the biggest effect in the regions with the most amount of background.

[6] The ideal readout time for one chip is $t = 917\,504\,\text{bits} \cdot 25\,\text{ns} = 22.9942\,\text{ms}$ [146], but this does not take into account overhead from the FPGA, sending data to the computer and processing in TOS. We will later see that the practical readout time of the final detector is closer to almost 500 ms under high rate conditions (e.g. ^{55}Fe calibration runs) and $\sim$ 200 ms for low rate background conditions.

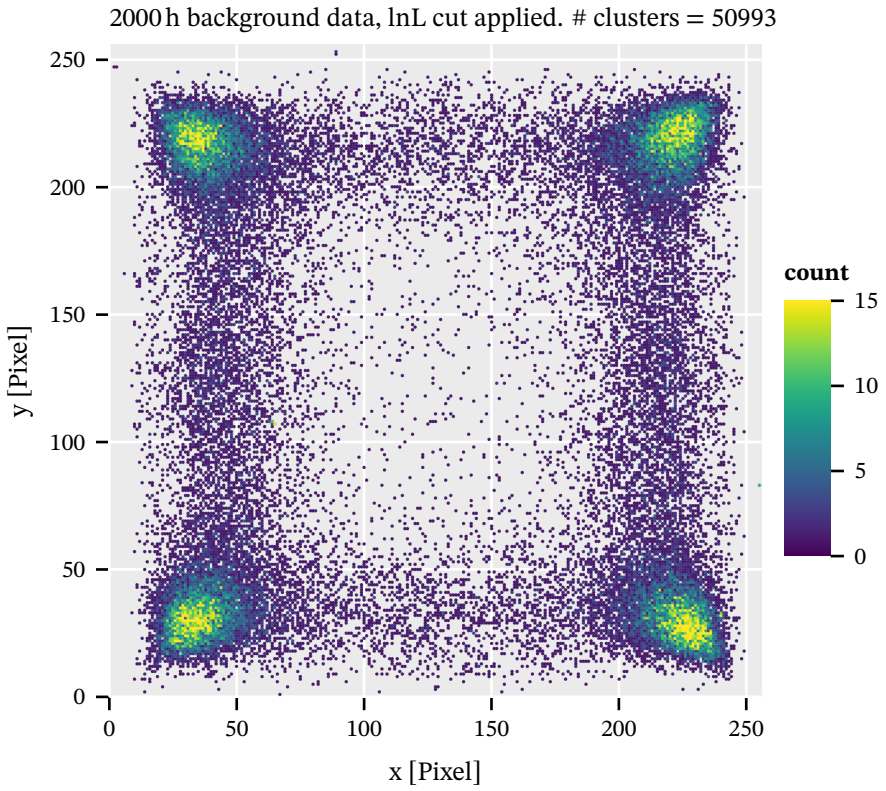


Figure 6.12.: Cluster centers left after likelihood cut applied to about 2000 h of background data. Background increasing dramatically towards edges and corners.

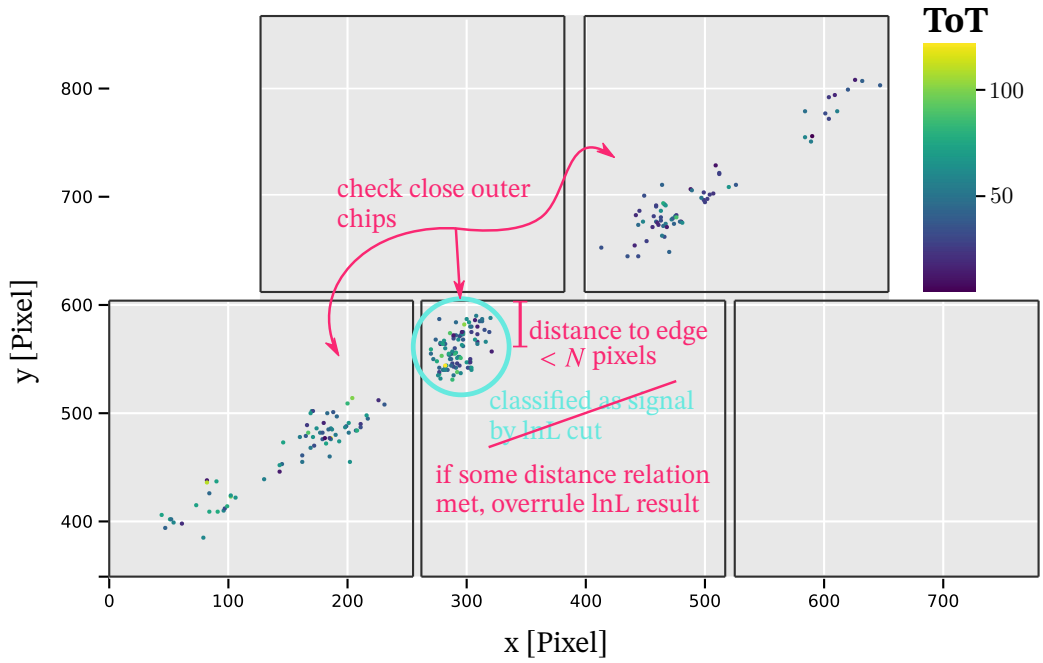


Figure 6.13.: Illustration of the basic idea behind the GridPix veto ring. If a cluster on the center chip is X-ray like and near the corners, checking the outer chips close to the corner for a track containing the center cluster can overrule the X-ray like definition of the center chip only.

6.11. Water cooling and temperature readout for the septemboard

During development of the septemboard one particular set of problems manifested. While testing a prototype board with 5 active GridPix in a gaseous detector, the readout was plagued by excessive noise problems. The detector exhibited a large number of frames with more than 4096 active pixels (the limit for a zero suppressed readout) and common pixel values of 11 810 indicating overrun ToT counters. On an occupancy (sum of all active pixels) of the individual chips, it is quite visible the data is clearly not due to cosmic background. Fig. 6.14 shows such an occupancy with the color scale topping out at the 80^{th} percentile of the counts for each chip individually. The chip in the bottom left shows a large number of sparks (overlapping half ellipses pointing downwards) at the top end. Especially the center chip in the top row shows highly structured activity, which is in contrast to the expectation of a homogeneous occupancy for a normal background run. In addition, on all chips some level of general noise on certain pixels is visible (some being clearly more active than others resulting in a scatter of 'points').

The intermediate board and carrier board used during these tests were the first boards equipped with two PT1000 temperature sensors. One on the bottom side of the carrier board and another on the intermediate board. Each is read out using a `MAX31685` micro controllers. Both of which are communicated with via a `MCP2210` USB-to-SPI micro controllers over a single USB port on the intermediate board. The single `MCP2210` communicates with both temperature sensors via the Serial Peripheral Interface (SPI) (see sec. A.2.4 for more information about the temperature logging and readout). In the run shown in fig. 6.14 the temperature sensors were not functional yet, as the readout software was not written. The required logic was added to the Timepix Operating Software (TOS), the readout software of the detector, motivated by this noise activity to monitor the temperature before and during a data taking period. The temperature on the carrier board indicated temperatures of $\sim 75\,°C$ in background runs similar to the one of fig. 6.14. One way to get a measure for the noise-like activity seen on the detector is to look at the rate of active pixels over time. With values well above numbers expected due to background, excess temperature seemed a possible cause for the issues. As no proper cooling mechanism was available, a regular desk fan was placed pointing at the detector when it was run without any kind of shielding. This saw the temperature under the carrier board drop from 76 °C down to 68 °C. As a result the majority of noise disappeared as can be seen in fig. 6.15(b) with the temperature curve during the full run in fig. 6.15(a). [7] [8]

The features visible in the occupancy plots are thus likely multiple different artifacts due to too

[7] See the full thesis version for the occupancy of the run with temperature readout in the subsection after this if interested.

[8] The realization that the issues are purely due to temperature effects was only after several months of eliminating many other options, both on the software as well as the hardware side. In particular power supply instabilities were long considered to be a source of problems. While they possibly also had an impact, better power supplies were built with larger capacitors to better deal with large variations in required power.

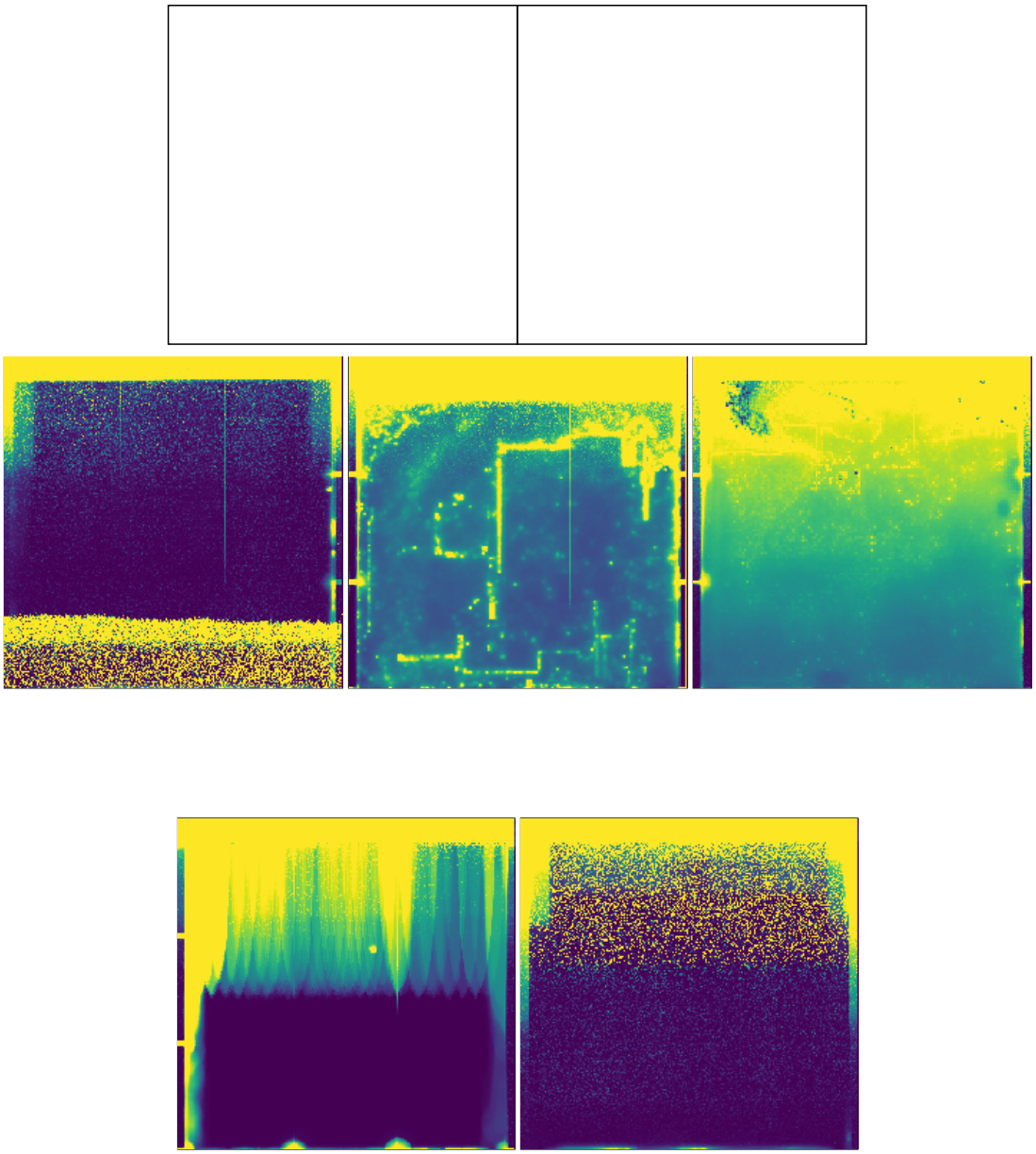
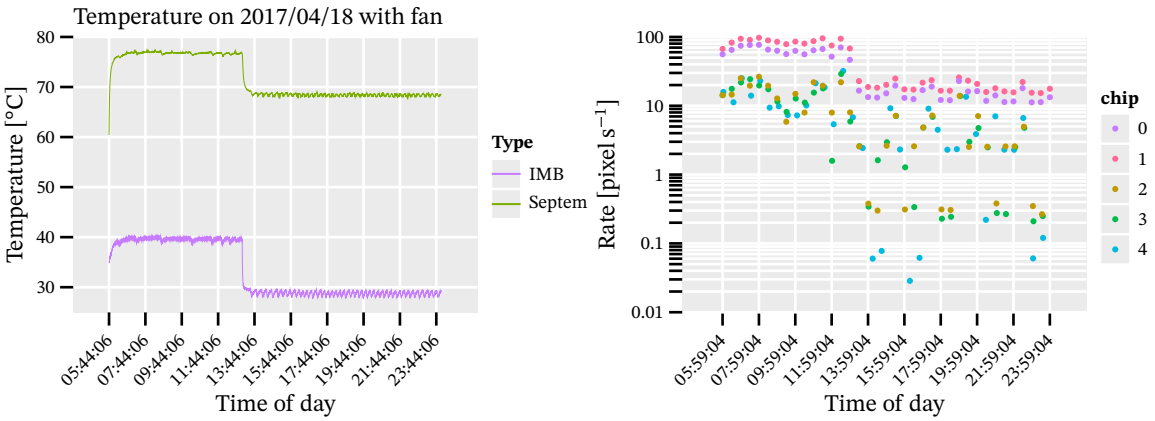


Figure 6.14.: Occupancy of a testing background run with $\mathcal{O}(1\,\mathrm{s})$ long frames using septemboard F during development without any kind of cooling. This also shows the layout of the full septemboard with realistic spacing.

high temperatures. A mixture of real sparks (bottom left chip in fig. 6.14) and possible instabilities that possibly affect voltages for the pixels (and thus change the thresholds of each pixel). As the temperature is measured on the bottom side of the carrier board, temperatures in the amplification region are likely higher.



(a) Temperature

(b) Mean hit rate

Figure 6.15.: (a) shows the temperature on the bottom side of the carrier board ('septem') and intermediate board ('IMB') during the background run. The point at which the desk fan is placed next to the detector is clearly visible by the 8 °C drop in temperature from about 76 °C to 68 °C. (b) shows the mean hit rate of each of the 5 chips installed on the carrier board at the time during the same run. The placement of the desk fan is easily visible as a reduction in mean rate on all chips.

Following this a bespoke water cooling was designed by T. Schiffer, made from oxygen-free copper with 3 mm channels for water to circulate through the copper body. [188] The body has the same diameter as the intermediate board and is installed right below. The water circulation is handled by an off-the-shelf pump and radiator from Alphacool [9] intended for water cooling setups for desktop computers. The pump manages a water flow rate of about $0.3\,\mathrm{L\,min^{-1}}$ through the 3 mm channels in the copper. In common operation the temperatures on the carrier board are between 45 to 50 °C and noise free operation is possible.

6.12. Detector efficiency

For the applications at CAST, the detector is filled with Ar / iC_4H_{10} : 97.7 %/2.3 % gas. Combined with its 300 nm Si_3N_4 window, the combined detection efficiency can be computed, if the 20 nm Al coating for the detector cathode is included by computing the product of the different efficiencies. The efficiency of the window and coating are the transmissions of X-rays at different energies for each material t_i. For the gas, the absorption probability of the gas a_i is needed. As such

[9] https://www.alphacool.com/

$$\varepsilon_{\text{tot}} = t_{\text{Si}_3\text{N}_4} \cdot t_{\text{Al}} \cdot a_{\text{Ar/iC}_4\text{H}_{10}}$$

describes the full detector efficiency assuming the parts of the detector, which are not obstructed by the window strongbacks. For a statistical measure of detection efficiency the occlusion of the window needs to be taken into account. Because it is position (and thus area) dependent, the need to include it is decided on a case by case basis. For the absorption of a gas mixture, we can use Dalton's law and compute the absorption of the individual gases according to their mole fractions (their percentage as indicated by the gas mixture) and then compute it for each partial pressure

$$a_i = \text{Absorption}(P_{\text{total}} \cdot f_i)$$

where P_{total} is the total pressure of the gas mixture (in this case 1050 mbar) and f_i is the fraction of the gas i. 'Absorption' simply refers to the generic function computing the absorption for a gas at a given pressure (see sec. 5.3.1 and sec. 5.1.1).

The full combined efficiency as presented here is shown in fig. 6.16. Different aspects dominate the combined efficiency (purple line) in different energy ranges. At energies above 5 keV the probability of X-rays to not generate a photoelectron within the 3 cm of drift distance becomes the major factor for a loss in efficiency. This means the combined efficiency at 10 keV is slightly below 30 %. The best combined efficiency of about 95 % is reached at about 3.75 keV where both the absorption is likely and the energy is high enough to transmit well through the window. The argon $K1s$ absorption edge is clearly visible at around 3.2 keV. At energies below the mean free path of X-rays is significantly longer as the $K1s$ absorption is a significant factor in the possible generation of a photoelectron. The window leads to a similar, but inverse, effect namely due to the $K1s$ line of Si at around 1.84 keV. Because transmission is desired through the window material, the efficiency *increases* once we go below that energy. Finally, the nitrogen $K1s$ line also contributes to an increase in efficiency once we cross below about 400 eV. The average efficiencies in the energy ranges between 0 to 3 keV and 0 to 10 keV are 73.42 % and 67.84 %, respectively.

The improvement in efficiency at energies below 3 keV in comparison to the mylar window used in the 2014/15 detector (see sec. 6.9) leads to a significant improvement in possible signal detection at those energies, which is especially important for searches with peak fluxes around 1 to 2 keV as is the case for the axion-electron coupling or a possible chameleon coupling.

6.13. Data acquisition and detector monitoring

The data acquisition software used for the Septemboard detector, the Timepix Operating Software (TOS), is not of direct importance for the thesis. But a longer section about it can be found in appendix

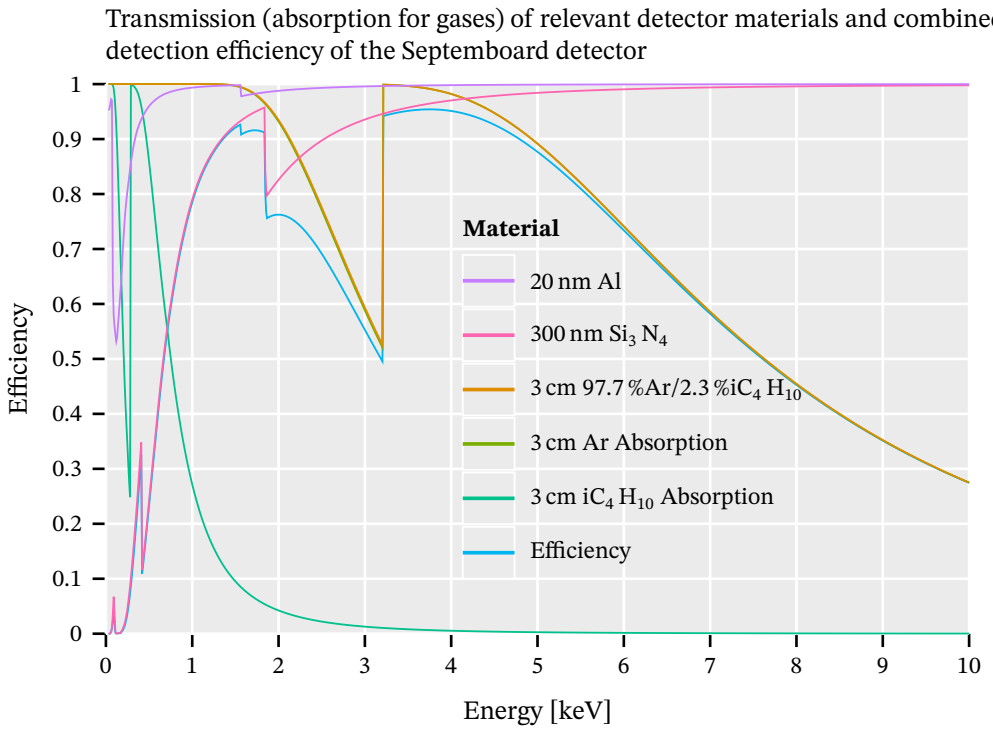


Figure 6.16.: Combined detection efficiency for the full detector, taking into account the gas filling of 1050 mbar Ar / iC_4H_{10}, the 300 nm Si_3N_4 window and its 20 nm Al coating.

A. It includes discussions about how the software works internally, how it is used, what the data format produced looks like and its configuration files. Further, it goes over the temperature readout functionality and finally presents the software used for detector monitoring, in the form of an event display used at CAST.

7. Detector calibration for operation Detector

Contents

This chapter introduces the calibrations required to get the Septemboard detector into a usable state for data taking at an experiment, in particular to interpret the data taken with it, sec. 7.1. Those calibrations purely related to the Timepix ASIC itself – to work noise free at the lowest possible thresholds – can be found in appendix C.1. Also the correct working parameters for the FADC are discussed in sec. 7.2. The scintillators need to be set to their correct discriminator thresholds, see sec. 7.3.

7.1. Timepix calibrations

As alluded to above, here we will focus on interpreting data taken with a Timepix ASIC. This means introducing the Time over Threshold (ToT) calibration method used to interpret ToT values as recorded charges, sec. 7.1.1. Further, based on recorded charge the gas gain can be determined. This is discussed in sec. 7.1.2.

Important references for the Timepix in general and for the calibration procedures explained below and in the appendix are [142, 145, 143, 146].

7.1.1. ToT calibration

The purpose of the ToT (**T**ime **o**ver **T**hreshold) calibration is not to perform a calibration for stable operation of a Timepix based detector, but rather to interpret the data received. It is needed to interpret the ToT values recorded by each pixel as a charge, i.e. a number of recorded electrons.

This is done by injecting charges onto the individual pixels – 'test pulses'. Capacitors are present to inject very precise voltage bursts onto the pixels. In case of the Timepix 1, each pixel uses a capacitance of 8 fF [143]. Knowing the capacitance and the voltage induced on them, the number of injected electrons can be easily calculated from

$$Q = CU.$$

By varying the injected charge and recording the resulting ToT values of the pixels, a relation between electrons and ToT values is determined:

$$f(p) = ap + b - \frac{c}{p - t}$$

where a, b, c and t are parameters to be determined via a numerical fit and p is the test pulse height in mV.

As such, inverting the relation this can be used to compute a charge for a given ToT value:

$$f(\mathrm{ToT}) = \frac{\alpha}{2a}\left(\mathrm{ToT} - (b - at) + \sqrt{(\mathrm{ToT} - (b - at))^2 + 4(abt + ac - at\mathrm{ToT})}\right)$$

where ToT is the time over threshold value recorded for a pixel, the constants a, b, c, t the fit parameters determined above and α the conversion factor relating the number of electrons from a pulse height of 1 mV.

An example of a ToT calibration of one chip is shown in fig. 7.1.

7.1.2. Pólya distribution & gas gain

In sec. 5.3.6 we introduced the Pólya distribution to describe the statistical distribution of the gas amplification stage. In the practical context of the Septemboard detector and the ToT calibration then, this is the histogram of all charge values recorded by the detector (and related of all ToT values). As the ToT calibration function is non-linear though, the histogram of the Pólya distribution has equal bin widths in ToT space, but increasing bin widths in charge space.

Such a histogram can be seen in fig. 7.2, for a 90 min slice of background data of the center chip of the Septemboard. The reasoning behind looking at a fixed time interval for the Pólya will be explained in section 10.2.2. The pink line represents the fit of the Pólya distribution following eq. 5.11 to the data. The dashed part of the line was not used for the fit and is only an extension using the final fit

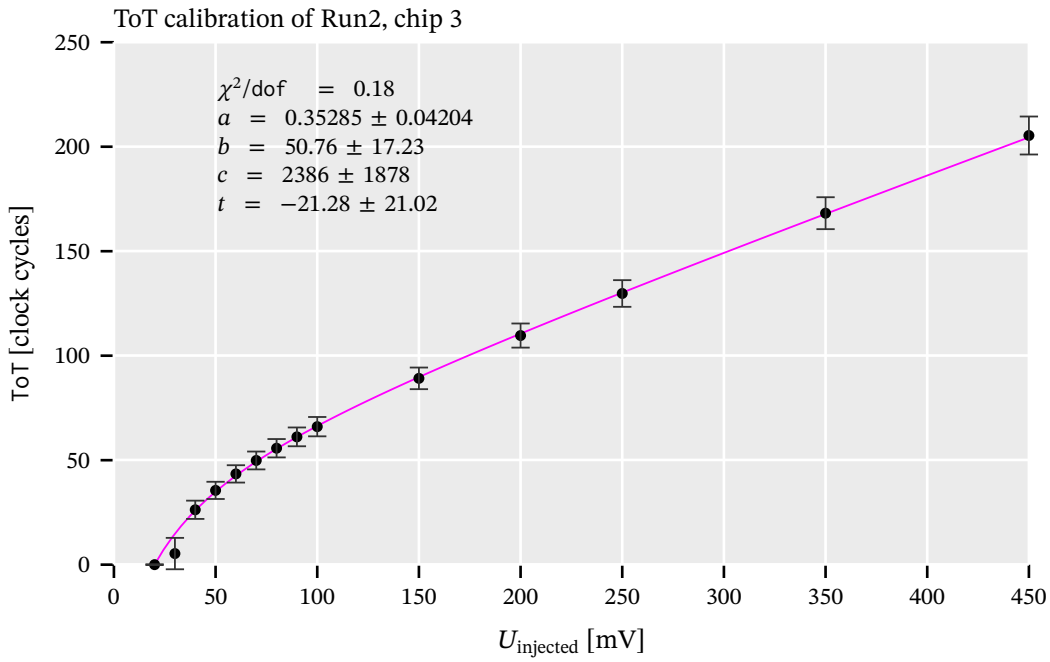


Figure 7.1.: Example of a ToT calibration measurement for the chip H10 W69, the center chip of the Septemboard, as it was done for the CAST data taking period 2.

parameters. At the lower end of charge values, a cutoff due to the chip's activation threshold is clearly visible. Note also how the bin widths increase from low to high charge values. The fit determines a gas gain of 3604.3, compared to the mean of the data yielding 3171.0. Following [134] any number for the gas gain used in the remainder of the thesis refers to the *mean of the data* and not the fit parameter. We use the fit mainly as a verification of the general behavior.

A secondary use case for the Pólya distribution is the determination of the activation threshold via the seen cutoff. More on this in appendix C.1.4.

7.1.3. Final Septemboard calibrations

The detector was calibrated according to the descriptions of the previous sections and appendix C prior to both major data taking campaigns at CAST (see sec. 9.6 for a detailed overview of both campaigns), once in October 2017 and then again in July 2018.

For an overview of all calibration parameters of interest, see the appendix C.2. Tables 7.1 and 7.2 show the THL and THS DAC [1] values used for the Septemboard detector at CAST during the data taking campaign from October 2017 to March 2018 (Run-2) and October 2018 to December 2018 (Run-3), respectively. The other DACs were all set to the same values for all chips in both data taking campaigns with the detector, shown in tab. 7.3.

[1] The THL DAC is the global threshold DAC of all pixels. The THS DAC is responsible for the range in which each pixel can be adjusted around the global value. See appendix C.1 for more information.

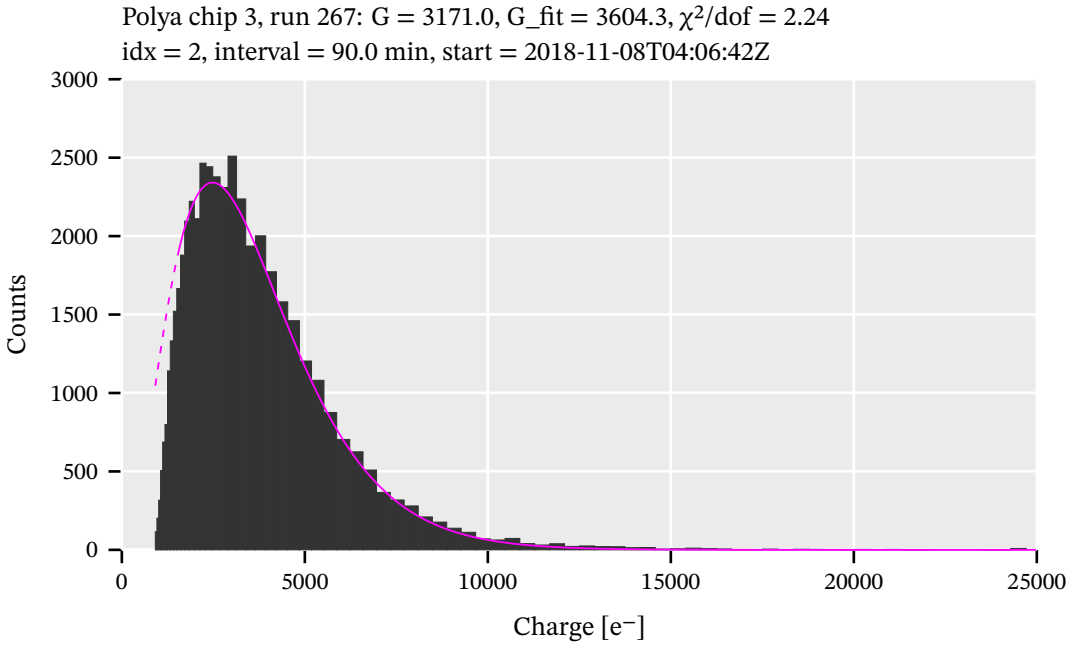


Figure 7.2.: An example of a Pólya distribution of chip 3 using the calibration of July 2018 based on 90 min of background data. A cutoff at low charges is visible. The pink line represents the fit of the Pólya distribution to the data. In the dashed region the line was extended using the final fit parameters.

DAC	E6 W69	K6 W69	H9 W69	H10 W69	G10 W69	D9 W69	L8 W69
THL	435	435	405	450	450	400	470
THS	66	69	66	64	66	65	66

Table 7.1.: The THL and THS DAC values for each of the chips of the Septemboard (board H) detector used at CAST for the data taking campaign from October 2017 to March 2018 (Run-2).

DAC	E6 W69	K6 W69	H9 W69	H10 W69	G10 W69	D9 W69	L8 W69
THL	419	386	369	434	439	402	462
THS	68	66	66	65	69	65	64

Table 7.2.: The THL and THS DAC values for each of the chips of the Septemboard (board H) detector used at CAST for the data taking campaign from October 2018 to December 2018 (Run-3).

DAC	Value
IKrum	20
Hist	0
GND	80
Coarse	7
CTPR	4294967295
BiasLVDS	128
SenseDAC	1
DACCode	6
RefLVDS	128
Vcas	130
ExtDAC	0
Disc	127
Preamp	255
FBK	128
BuffAnalogA	127
BuffAnalogB	127

Table 7.3.: DAC values and settings that are common between data taking periods and all chips of the Septemboard for CAST, from the `fsr` configuration file.

7.1.4. High voltage

The high voltage (HV) settings used for the Septemboard detector are shown in tab. 7.4. The target is a drift field on the order of $500\,\mathrm{V\,cm^{-1}}$ and an amplification field of about $60\,\mathrm{kV\,cm^{-1}}$. The main voltages to choose are the grid voltage (to determine the amplification field) and the cathode voltage (to determine the drift field). The other voltages are computed based on a constant field gradient. Entries ring 1 and ring 29 are the voltages applied to the field shaping ring running around the detector volume to achieve a more homogeneous field. The HV for the Septemboard is controlled via an iseg [2] HV module, while the veto scintillator (requiring positive high voltage) is controlled via a CAEN N470 [3].

Description	Channel	Voltage / V	TripCurrent / mA
grid	0	-300	0.050
anode	1	-375	0.050
cathode	2	-1875	0.050
ring 1	3	-415	0.100
ring 29	4	-1830	0.100
veto scinti	5	+1200	2
SiPM	6	-65.6	0.05

Table 7.4.: Table of high voltages in use for the InGrid Mk. IV. Note that the veto scintillator is not controlled via the iseg module, but by a CAEN N470.

7.2. FADC calibration

The FADC requires care about multiple aspects. First, the settings need to be chosen that configure both the operating characteristics, data readout and trigger threshold. Next, the Ortec 474 shaping amplifier has multiple different settings. Finally, in order to interpret the signals received from the FADC, a so-called ”pedestal run” should be recorded.

FADC settings The FADC settings – more details in the configuration file explanation of appendix section A.2.2 – configure the FADC to run at a frequency of 1 GHz as to cover a running time interval of 2.56 μs in each channel. While it could run at up to 2 GHz, 1 ns per time bin is accurate enough given the time scales associated with the amplifier (see below) and a longer interval is more useful. Further, an external trigger is sent out to TOS if the threshold is exceeded. The

[2] https://iseg-hv.com/
[3] https://caen.it/

threshold itself is set to −40 mV [4]. The value was chosen based on trial and error to avoid no triggers based on baseline noise. Given the range of up to −1 V, the relative threshold is pretty low. Finally, the operation mode and data readout is set, the input channel is chosen and the pedestal run parameters are configured (see below).

Amplifier settings The 474 Ortec shaping amplifier has 3 settings of note. The absolute gain as a multiplier and a signal integration as well as differentiation time. The initial settings were set to an amplification of 6x, an integration time of 50 ns and a differentiation time of 50 ns. However, these were changed during the data taking campaign, see more on this in section 10.4.1.

Pedestals The $4 \cdot 2560$ registers of the FADC are part of 4 separate cyclic registers. Due to hardware implementation details, the absolute recorded values of each register is arbitrary. In a pedestal run multiple measurements, $\mathcal{O}(10)$ of a certain length (100 ms in our case), are performed and the pedestal values averaged. The resulting values represent a mean value for the typical value in each register, hence the name 'pedestal'. To interpret a real measured signal, these pedestals are subtracted from the recorded signal. Each of the 4 FADC channels may have very different pedestal values, but within a single channel they are usually within $\lesssim$ 50 ADC values. Fig. 7.3 shows the pedestals of all 4 FADC channels as they were recorded before CAST data taking started. The pedestals drift over time, but the associated time scales are long. Alternatively, the pedestals can be computed from real data by computing a truncated mean in each register, which we'll discuss later in sec. 8.5.1.

More details to each of these will be given later where it is of importance.

7.3. Scintillator calibration

The final piece of the detector requiring calibration, are the two scintillators. As both of them are only used as digital triggers and no analogue signal information is recorded, a suitable discriminator threshold voltage has to be set.

7.3.1. Large scintillator paddle

The large veto scintillator paddle was calibrated at the RD51 laboratory at CERN prior to the data taking campaign in March 2017. Using two smaller, calibrated scintillators to create a coincidence

[4] The trigger threshold DAC is a 12-bit DAC. Its values correspond to −1 V at `000` and 1 V at `FFF`. Hence 1966 (seen in the configuration file) is roughly −40 mV.

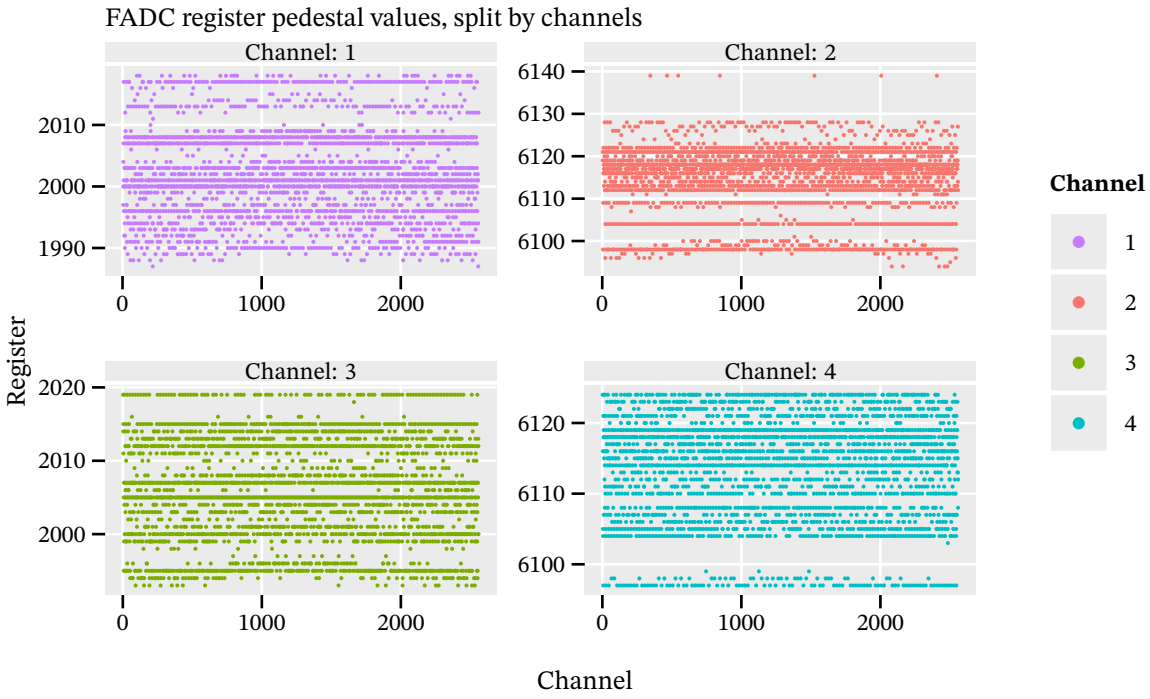


Figure 7.3.: The FADC pedestal run used for CAST data initially, split by each of the 4 FADC channels. Each channel's pedestals vary by about $\mathcal{O}(30)$ ADC values. The first few registers in each channel are not shown, as they are outlying by ~ 100.

setup of the three scintillators, measurements were taken at different thresholds. Each measurement was 10 min long. An amplifier was installed after the PMT to increase the signal.

The Canberra 2007 base and Bicron Corp. 31.49x15.74M2BC408/2-X PMT require a positive high voltage. It was supplied with 1200 V. Table 7.5 shows the recorded measurements. Based on these a threshold of −110 mV was chosen for the CAST data taking. Fig. 7.4 also shows the data from table. While the coincidence counts at −110 mV are lower than the visible plateau starting at −100 mV, this threshold was chosen, because the raw counts were still considered too high compared to expectation based on cosmic muon rate and the size of the scintillator. [5]

Threshold / mV	Counts Szinti	Counts Coincidence
-59.8	31221	634
-70.0	30132	674
-80.4	28893	635
-90.3	28076	644
-100.5	27012	684
-110.3	25259	566
-120.0	22483	495
-130.3	19314	437
-140.3	16392	356
-150.5	13677	312
-160.0	11866	267
-170.1	10008	243

Table 7.5.: Measurements for the calibration of the large veto scintillator taken at RD51 at CERN with two smaller, calibrated scintillators in a coincidence. Each measurement was 10 min. The thresholds set on the discriminator for the veto scintillator were originally measured with a 10x scaling and have been rescaled here to their correct values.

7.3.2. SiPM

The SiPM was calibrated during the bachelor thesis of Jannes Schmitz in 2016 [196] based on a coincidence measurement with calibrated scintillators.

[5] While it is unclear to me now given it's been over 5 years, I believe at the time of the calibration we wrongly assumed a muon rate of 100 Hz m^{-2} sr^{-1} instead of about 1 cm^{-2} min^{-1}. The former number only works out if one integrates it over the $\cos^2(\theta)$ dependence, but only along θ and not φ! Either way, the number seems problematic. However, it did misguide us in likely choosing a too low threshold, as using the former number yields an expected number of counts of $\sim$ 32 000 compared to only $\sim$ 20 000 in our naive approach.

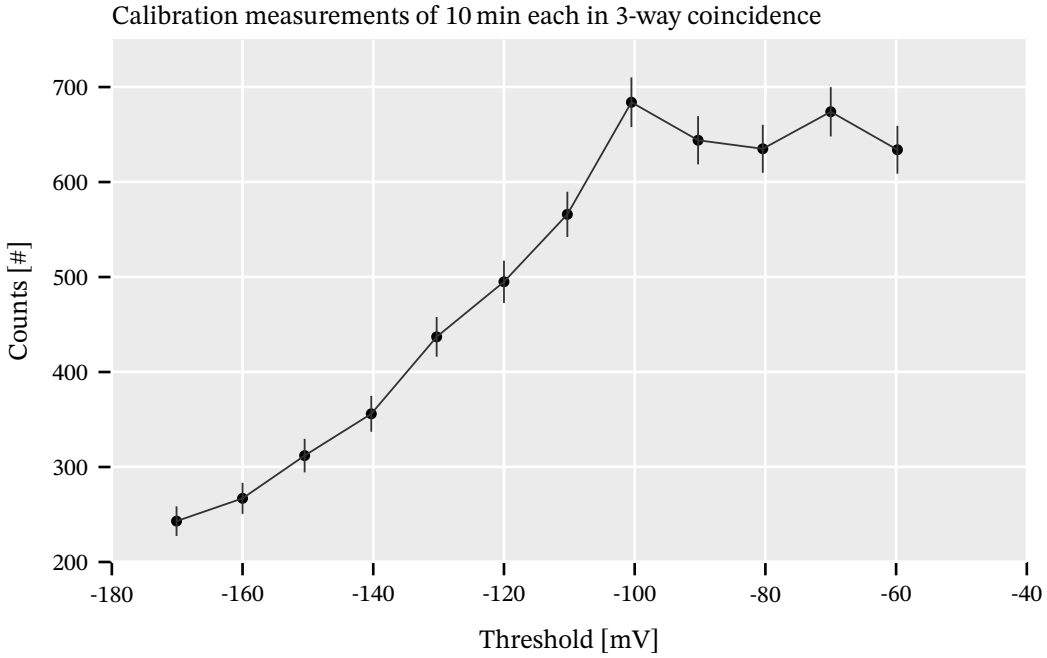


Figure 7.4.: Calibration measurements for the veto scintillator printed in table 7.5. The line is an interconnection of all data points. The errors represent Poisson-like $\sqrt{N}$ uncertainties.

8. Data reconstruction Reconstruction

Contents

We will now go through the general data reconstruction for data taken with the Septemboard detector. Starting with a short introduction to the data analysis framework `TimepixAnalysis` developed for this thesis, sec. 8.1. Then we cover the data parsing of the ASCII based data format produced by TOS (see appendix A.2.1 for a description of the data format) in sec. 8.2. At this point we shortly discuss our expectation of the data properties recorded with the detector, sec. 8.3, as it motivates the kind of reconstruction that is performed. From here the explanation of the data reconstruction starts, sec. 8.4, including cluster finding (sec. 8.4.1) and calculation of geometric properties, sec. 8.4.2.

The FADC data reconstruction follows in section 8.5. Finally, the scintillators are mentioned in section 8.6.

There is an additional long section in the appendix O that goes through the software used for the data reconstruction intended for people using these tools in the future. And appendix P shows how the full data reconstruction for all CAST data presented in this thesis is performed.

8.1. TimepixAnalysis and Nim

The data reconstruction software handling the processes mentioned in the remainder of this chapter is the TimepixAnalysis [194] [1] framework. It is only a "framework" in a loose sense, as it is a set of programs to parse data from different Timepix (usually GridPix) DAQ software packages, process and analyze it. In addition, it contains a large number of tools to visualize that data as well as analyze auxiliary data like lists of data taking runs, CAST log files and more.

The entire code base is written in the Nim programming language [185] [2]. Nim is a statically typed, compiled language with a Python-like whitespace sensitive syntax, taking inspirations from Pascal, Modula and in particular ideas regarding type safety from Ada. Further, it has a strong metaprogramming focus with full access to its abstract syntax tree (AST) at compile time, offering Lisp-like macro functionality. This allows to construct powerful and concise domain specific languages (DSLs). Nim compiles its code by default first to C code, which can then utilize the decades of compiler optimization techniques available via GCC [93] or Clang [138], while allowing to target every single platform supported by C (which effectively means almost all). As such it also achieves performance on par with C, while providing high-level features normally associated with languages like Python.

Nim was selected as the language of choice for TimepixAnalysis, due to its combination of concise and clean syntax, high-level features for productivity, high performance due to native code generation, easy interfacing with existing C and C++ code bases and its strong metaprogramming features, which allow to reduce boilerplate code to a minimum.

Appendix O.1 contains a detailed overview of the important tools part of TimepixAnalysis and how they are used for the context of the data analysis presented in this thesis. Read it if you wish to understand how to recreate the results presented in this thesis.

8.2. TOS data parsing

The first part of the GridPix data reconstruction is the parsing of the raw ASCII data files, presented in A.2.1. This is implemented in the raw_data_manipulation program [3], part of TimepixAnalysis [194]. Its main purpose is the conversion of the inefficient ASCII data format to the more appropriate and easier to work with HDF5 [4] [214] format, a binary file format intended for scientific datasets. While this data conversion is the main purpose, pixels with ToT or ToA values outside a user defined

[1] https://github.com/Vindaar/TimepixAnalysis

[2] https://nim-lang.org

[3] https://github.com/Vindaar/TimepixAnalysis/blob/master/Analysis/ingrid/raw_data_manipulation.nim

[4] HDF5 [214] is the Hierarchical Data Format of version 5. It is a binary data format intended for scientific datasets, which uses an in-file layout similar to a virtual file system. Datasets (equivalent to files) are stored in groups (equivalent to directories). Metadata can be attached to either and linking between datasets, even across files is supported. It supports a large number of compression filters to reduce the file size of the stored data.

range can be filtered out at this stage [5]. Each data taking run is processed separately and will be represented by one group (similar to a directory on a file system) in the output HDF5 file. See appendix section O.1.4 for an explanation of the produced data layout.

As the data is being processed anyway, at this time we already compute an occupancy map for each chip in the run. This allows for a quick glance at the (otherwise unprocessed) data.

8.3. Expectation of event shapes

Based on the theoretical aspects of a gaseous detector as explained in chapter 5 and the expected kinds of signal sources at an experiment like CAST (see chapter 9), we can have a good expectation of the kinds of signals that a GridPix detector records for different types of events.

The signal source we are interested in for an axion helioscope are soft energy X-rays, below 10 keV. The main goal in later determining a background rate and computing a physics result from data is to filter out these X-rays from the rest of the data the detector records. The dominant source of background in any gaseous detector at surface level is due to cosmic muons.

Fortunately, muons and X-rays behave very different in the detector. X-rays generally produce a single photoelectron, which creates further primary electrons in a local region. These drift under transverse diffusion to the readout plane, which effectively gives them a roughly circular shape. Muons on the other hand produce electrons (which each produce further local primaries) on a track along their entire path through the gas volume. Under most angles this implies their shape is very eccentric, i.e. 'track-like'.

Two example events, one of a $\sim 5.9\,\text{keV}$ ^{55}Fe X-ray and the other of a typical muon is shown in fig. 8.1.

Given the distinct geometric properties of these different types of events and the fact that a GridPix provides extremely high spatial resolution and single electron efficiency, the data reconstruction fully embraces this. Most of the computed properties, which we will introduce in the next sections, are directly related to geometric properties of the events.

8.4. Data reconstruction

With the data stored in an HDF5 file after processing the raw data with `raw_data_manipulation`, the actual data and event reconstruction can begin. This is handled by the `reconstruction` [6] program.

[5] These type of cuts are applied at this stage of the processing, because for certain use cases or certain detectors specific `ToT` or `ToA` ranges are of no interest / contain junk data (because of a faulty chip for example). In this case it is useful to remove such data in this preprocessing stage to lighten the workload for anything after.

[6] `https://github.com/Vindaar/TimepixAnalysis/blob/master/Analysis/ingrid/reconstruction.nim`

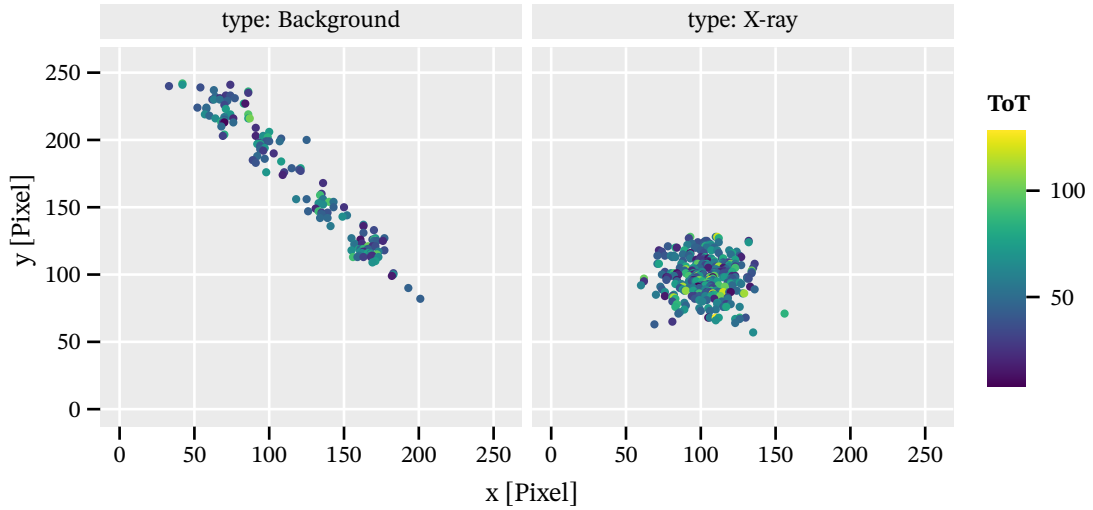


Figure 8.1.: Two example events one might see in the detector, left a common background event of a (likely) muon track, which enters the readout plane (hence the slightly triangular shape) and right a classical 5.9 keV X-ray from a ^{55}Fe calibration source.

It continues from the HDF5 file created before and proceeds to reconstruct all runs in the given input file.

For each run, each GridPix chip is processed sequentially, while all events for that chip are then processed in parallel using multithreading. For each event, the data processing is essentially a two step process:

1. perform cluster finding, see section 8.4.1.
2. compute geometric properties for each found cluster, see section 8.4.2.

8.4.1. Cluster finding

The cluster finding algorithm splits a single event into possibly multiple clusters. Clusters are defined based on a certain notion of distance (the details depend on the clustering algorithm used). The multiple clusters from a single event are then treated fully equally for the rest of the analysis. The fact that they originate from the same event has no further relevance (with a slight exception for one veto technique, which utilizes clustering over multiple chips, more on that in section 11.5.3).

There are two different cluster finding algorithms implemented for use in `TimepixAnalysis`. The default one is strictly used for the general cluster finding as part of the reconstruction, the other is intended to be used for one of the vetoes (again, sec. 11.5.3). The choice is user configurable however. [7]

[7] The clustering logic of TPA is found here: `https://github.com/Vindaar/TimepixAnalysis/blob/master/Analysis/ingrid/private/clustering.nim`

Default The default one is the same clustering algorithm introduced for the data reconstruction of the 2014/15 GridPix detector in [134]. It defines a cluster by all pixels within the squares of side length N centered around each pixel. It is best thought of as a recursive square neighbor search around each pixel. For each neighbor in the search square, start another search square. Once no neighbor finds any neighbors not already part of the cluster, it is finished.

DBSCAN The secondary clustering algorithm is the **D**ensity-**b**ased **s**patial **c**lustering of **a**pplications with **n**oise (DBSCAN) [83] algorithm. In contrast to the default algorithm it is – as the name implies – a density based algorithm. This means it distinguishes points which have more neighbors (high density) from those with few neighbors (low density). The algorithm has a parameter `minSamples`, which defines the density threshold. If a point has at least `minSamples` neighbors within a (euclidean) distance of ε (the second parameter) it is considered a "core point". All core points build a cluster with all other points in their reach. Those points in reach of a core point, but do itself not have `minSamples` neighbors are still part of the cluster. Any point not in reach of a core point is a "noise point". The main advantage of this algorithm over many other more classical algorithms is the ability to separate clusters close to one another, which are not separateable by a linear cut. This results in a more humanly "intuitive" clustering. DBSCAN is one of the most widely used clustering algorithm in many scientific fields and even in 2017 was still considered highly relevant [197].

Another clustering algorithm (currently not implemented) is CLASSIX [60], which promises fast clustering based on sorting along the first principal component axis. Based on its properties as presented in its paper it could be an extremely useful algorithm for our application and should be investigated in the future.

8.4.2. Calculation of geometric properties

For each individual cluster the geometric event reconstruction is up next. As the basic differentiator between X-rays and common background events is their circularity, most properties are in some sense related to how eccentric clusters are. Therefore, the first thing to be computed for each cluster, is the rotation angle [8].

The rotation angle is found via a non linear optimization of

[8] Note that the absolute value of the rotation angle is of secondary importance. For X-rays the rotation angle is going to be random, as the definition of a long and short axis in a (theoretically perfect) circle depends on the statistical distribution of the pixels. However, for pure muons it allows to map the rotation angle to the incidence angle.

$$x_i' = \cos(\theta)\,(x_i - \bar{x}) \cdot P - \sin(\theta)\,(y_i - \bar{y}) \cdot P$$
$$y_i' = \sin(\theta)\,(x_i - \bar{x}) \cdot P + \cos(\theta)\,(y_i - \bar{y}) \cdot P$$

where θ is the rotation angle (in the context of the optimization the parameter to be fitted), x_i, y_i the coordinates of the i-th pixel in the cluster, and $\bar{x}, \bar{y}$ the center coordinates of the cluster. $P = 55\,\mu\text{m}$ is the pixel pitch of a Timepix. The resulting variables x_i', y_i' define a new rotated coordinate system. From these coordinates, the RMS [9] of each of these new axes is computed via

$$x_{\text{RMS}} = \sqrt{\frac{1}{N}\left(\sum_i x_i'^2\right) - \frac{1}{N^2}\left(\sum_i x_i'\right)^2}$$
$$y_{\text{RMS}} = \sqrt{\frac{1}{N}\left(\sum_i y_i'^2\right) - \frac{1}{N^2}\left(\sum_i y_i'\right)^2}.$$

Based on these we then simply redefine

$$\sigma_{\text{transverse}} = \min(x_{\text{RMS}}, y_{\text{RMS}})$$
$$\sigma_{\text{longitudinal}} = \max(x_{\text{RMS}}, y_{\text{RMS}}),$$

which then define the eccentricity ε to (see also fig. 8.2(b))

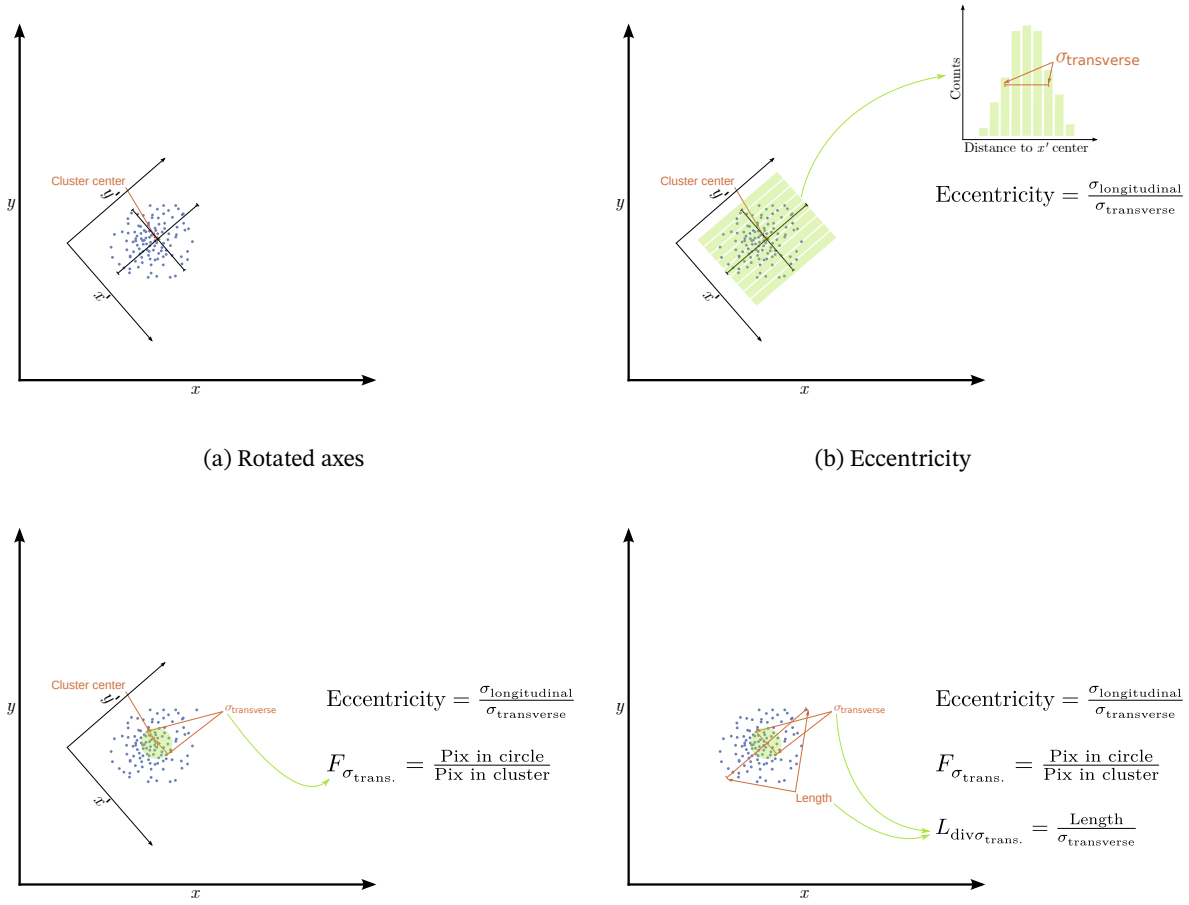
$$\varepsilon = \frac{\sigma_{\text{longitudinal}}}{\sigma_{\text{transverse}}},$$

guaranteeing $\varepsilon \geq 1$.

During the non linear optimization, the algorithm attempts to maximize the eccentricity. In a track like cluster, the maximum eccentricity is found under the rotation angle θ, which points along the longest axis of the cluster. The resulting rotated coordinate system after the fit has converged, is illustrated in fig. 8.2(a).

Once the rotation angle and therefore the rotated coordinate system of a cluster is defined, most other properties follow in a straight forward fashion. In the rotated coordinate system the axis along the long axis of the cluster is called ”longitudinal” and the short axis ”transverse” in the following. The higher moments skewness and kurtosis for each axis are computed as well as the length and width of the cluster based on the biggest spread of pixels along each axis. In addition to the geometric properties a few other properties like the number of pixels are also computed. Three of the most important variables are illustrated in fig. 8.2. These enter the likelihood cut definition as we will see in sec. 11.1.

[9] The term ’root mean square’ is used although we actually refer to the standard deviation of the sample. We follow [134], but this ambiguity is often encountered unfortunately.



(a) Rotated axes

(b) Eccentricity

(c) Fraction in transverse RMS

(d) Length divided by transverse RMS

Figure 8.2.: Schematic explanation of the basic cluster reconstruction and the three most important geometric properties. (a) defines the rotated coordinate system found by non-linear optimization of the long and short cluster axis. Along the long and short axes, (b), the transverse standard deviation $\sigma_{\text{transverse}}$ is computed, which then defines the eccentricity by this ratio. (c) shows the definition of a less obvious variable: the fraction of pixels within a circle of one $\sigma_{\text{transverse}}$ radius around the cluster center. Similarly (d) shows the full cluster length defined by the furthest active pixels in the cluster divided by $\sigma_{\text{transverse}}$ as another variable. These three variables enter the likelihood cut used for background suppression.

The following is a list of all properties of a single cluster computed by the `reconstruction` tool. The `ig` prefix is due to the internal naming convention. All but the likelihood, charge and energy properties are computed during the first pass of the tool, namely in the context discussed above. [10]

igEventNumber The event number the cluster is part of (multiple clusters may share the same event number).

igHits The number of pixels in the cluster.

igCenterX / igCenterY The center position of the cluster along the `x` / `y` axis of the detector.

igRotationAngle The rotation angle of the long axis of the cluster over the chip coordinate system.

igLength The length of the cluster along the long axis in the rotated coordinate system, defined by the furthest pixel at each end in that direction.

igWidth The equivalent of **igLength** for the short axis.

igRmsLongitudinal The root mean square (RMS) along the long axis.

igRmsTransverse The RMS along the short axis.

igSkewnessLongitudinal / igKurtosisLongitudinal The skewness / kurtosis along the long axis.

igSkewnessTransverse / igKurtosisTransverse The skewness / kurtosis along the short axis.

igEccentricity The eccentricity of the cluster, defined by the ratio of the longitudinal RMS over the transverse RMS.

igLengthDivRmsTrans The length of the cluster divided by the transverse RMS (see fig. 8.2(d)).

igFractionInTransverseRms The fraction of all pixels within a radius of the transverse RMS around the center (see fig. 8.2(c)).

igTotalCharge The sum of the charge of the `ToT` calibrated charges of all pixels in the cluster (see sec. 8.4.3).

igEnergyFromCharge The calibrated energy of the cluster in keV (see sec. 10.1).

igLikelihood The likelihood value of the cluster for the likelihood cut method, explained in detail in section 11.1.

[10] In particular all these properties are computed here: `https://github.com/Vindaar/TimepixAnalysis/blob/master/Analysis/ingrid/private/geometry.nim#L331-L391`

After the calculation of all geometric properties for all events and chips, the data is written to an output HDF5 file (similar in format to the output of `raw_data_manipulation`) for each run. This concludes the first pass of `reconstruction` over the data. See appendix section O.1.5 for an explanation of the produced data layout.

8.4.3. Data calibration

The next step of the reconstruction is the data calibration. This is a separate pass over the data as it is optional on the one hand and requires further inputs about each used GridPix than just the raw data (different calibration files) on the other hand.

There are different calibrations to be performed:

1. the charge calibration via the application of the `ToT` calibration as introduced in section 7.1.1.
2. the calculation of the gas gain, introduced previously in section 7.1.2 and more in sec. 10.2.2.
3. the energy calibration (see sec. 10.1 and 10.3).

The `ToT` calibration is in principle performed simply by converting each `ToT` value to an equivalent charge in electrons using the calibration as presented in section 7.1.1. For each GridPix used in a detector, a `ToT` calibration must be available.

`TimepixAnalysis` comes with a library and helper program, which manages a simple database about different GridPixes, their calibrations and their validity (in time and runs they apply to). The user needs to add the chips for which they wish to perform a `ToT` calibration to the database before it can be performed. See appendix O.1.10 for a detailed overview. For any chip part of the database, the `ToT` calibration is a single pass over the `ToT` values of all runs. This generates a calibrated charge for every pixel of every cluster and a combined property, the `totalCharge` of the full charge of each cluster.

Gas gain values are computed in 90 min time intervals for each chip. This strikes a good balance between enough statistics and reduced sensitivity to variation in gas gain due to external effects. As this deserves its own discussion, more on this in sec. 10.2.2.

Finally, while the energy calibration is also handled by `reconstruction`, we will cover it in section 10, due to its more complex nature.

8.4.4. Event duration

During the reconstruction of the data, another important parameter is computed, namely the event duration of each individual event. In principle each event has a fixed length, because the Timepix uses a shutter based readout, with the shutter length predefined. However, as the FADC is used as

an external trigger to close the shutter early, if it recorded a signal, all events with an FADC trigger have a shorter duration.

For the fixed length duration events their length is computed by the shutter length as indicated in TOS. In appendix A.2.1, listing 2 the `shutterTime` and `shutterMode` fields are listed. These define the absolute length of the shutter opening in (effectively) number of clock cycles. The `shutterMode` acts as a modifier to the number of clock cycles:

$$t_{\text{clocks}}(\texttt{mode}, t) = 256^{\texttt{mode}} \cdot t$$

where t is the `shutterTime` and `mode` corresponds to the `shutterMode`. The available modes are:

- `short`: 0
- `long`: 1
- `verylong`: 2

In case of the FADC triggering, the clock cycles after shutter opening that were recorded up to the trigger is also reported in the data files, see appendix sec. A.2.1, listing 3. With the number of clock cycles the shutter was open, the total event duration can then be computed in either case via:

$$d(t_{\text{clocks}}) = \frac{t_{\text{clocks}} \cdot 46}{40 \cdot 1\,000\,000}.$$

8.5. FADC reconstruction

The data files created from the FADC data sent upon a trigger are essentially memory snapshots of the circular register of the FADC. We will go through the necessary steps to convert that raw data into usable signals, given the FADC settings we use and the data TOS generates from it. See appendix sec. A.2.1 for an overview of the raw FADC data files. For a detailed overview of the FADC readout process see the FADC manual [55] [11].

In `TimepixAnalysis` FADC data is automatically parsed from the ASCII data files into HDF5 files as part of `raw_data_manipulation` if FADC files are present. The spectrum reconstruction is done automatically as part of the `reconstruction` program, but calculation of the baseline, rise and fall time is an optional step.

[11] A PDF of the FADC manual is available here: `https://archive.org/details/manualzilla-id-5646050/`

8.5.1. FADC pedestal calculation

As alluded to in sec. 7.2 the pedestal values cannot only be taken from a pedestal run recorded before data taking, but can also be extracted from real data, under the condition that a decent fraction of FADC registers in a single FADC event is on the baseline and normally distributed between events.

The idea is to look at an ensemble of values for each register taken from different events and remove all those events in each register, in which it was involved in a real signal. Due to the cyclic nature of the FADC registers, different registers will capture signals in each event. At least in typical signals recorded with a GridPix the signal lengths are $\mathcal{O}(10\,\%)$ of the window length, leaving plenty of registers free to recover pedestal information. Regular noise affects things, but is partially taken care by the truncation and partially cancels out as real noise is normal distributed around the actual pedestal. This latter approach is the one used in the data analysis by calculating:

$$p_i(r) = \text{mean}_{20^{\text{th}}}^{98^{\text{th}}}(\{r_i\}) \tag{8.1}$$

where $p_i(r)$ is the pedestal in register r_i and the mean is taken over all data $\{r_i\}$ in that register within the 20-th and 98-th percentile. All data refers to a full data run of ~ 1 day. The highly biased nature is due to the real signals being negative. Removing the smallest 20 % of data guarantees in the vast majority of events the full physical signal is excluded given the typical signal lengths involved. A small upper percentile is used to exclude possible significant outliers to the top. While such a biased estimator will not result in the real mean (and in case of signal and noise free input data thus the real pedestals), a slight bias is irrelevant, as the baseline is still calculated for each reconstructed signal which is used to correct any global offset.

8.5.2. FADC spectrum reconstruction

The first step to reconstruct the FADC signals, is to perform the pedestal correction. This is simply done by subtracting the pedestals register by register from the data file

$$N_{i,\text{corr}} = N_{i,\text{raw}} - N_{i,\text{pedestal}}$$

with the raw data $N_{i,\text{raw}}$ and the pedestals $N_{i,\text{pedestal}}$ in register i (as computed according to eq. 8.1).

With the pedestals removed, the temporal correction is next to unfold the data into the correct order. This needs to be performed on each of the 2560 registers for each channel separately. The temporal rotation is performed by shifting all registers by

$$n_{\text{rot}} = (\texttt{TRIG\_REC} - \texttt{POSTTRIG}) \cdot 20$$

places to the left. The constants TRIG_REC and POSTTRIG are from appendix section A.2.1, written in each data file in the header.

The final step is to convert the ADC values of each register into voltages in V. Given that the ADC covers the range of −1 to 1 V as the ADC values 0 to 4096 (16384) with 12 (14) bit [12], this means the conversion from ADC to volts is simply

$$U_i = \frac{N_{i,\text{corr}} - 2048}{2048}$$

when using the 12 bit operating mode for each register.

With these corrections applied, the recorded FADC spectrum is recovered, centered around the trigger position.

8.5.3. Signal baseline, rise time and fall time

Assuming a singular event is recorded with the FADC, the main properties of interest of the resulting signal pulse are the signal baseline and based on that the rise and fall time.

Computing the position of the baseline is generally a non trivial problem, as a priori the position, width and number of signals in the spectrum is unknown. A reasonable expectation though is that the majority of points in a signal should lie close to the baseline, as the fraction of the FADC window covered by a signal is typically less than a quarter. As such a somewhat empirical way to compute the baseline B was chosen using a biased truncated mean

$$B = \text{mean}_{30^{\text{th}}}^{95^{\text{th}}}(S)$$

between the 30^{th} and 95^{th} percentile of the data. The bias is intended to remove the impact of the negative signal amplitude and remove the worst positive outliers. An optimal solution would perform a rigorous peak finding for a signal pulse, remove those points and compute the mean of the remainder of the points. [13]

Once the baseline is defined it can be used to determine both the rise and the fall time. These are generally computed based on the number of registers between the minimum of the signal and some threshold slightly below the baseline (compare with fig. 8.3) in order to reduce the effect of local noise variations. While configurable, the default value of the threshold c_B (B for baseline) is

[12] The FADC can be operated in a 12 or 14-bit mode. We run in the 12-bit mode. Also see A.2.1.

[13] Aside from performing peak fitting (which is difficult and requires understanding of the expected signal shapes) another approach might be a local linear smoothing (e.g. a Savitzky-Golay filter with polynomial of order 1) in a suitable window range. The result would be a much more stable spectrum. This could then be used to compute the numerical derivative from which all those intervals with a slope smaller than some epsilon provide the dataset from which to compute the mean. The tricky aspect would be choice of window size and the behavior in very noisy events.

$$c_B = B - 0.1 \cdot |B - \min(S)|\,,$$

10 % of the difference between the baseline B and the minimum value of the spectrum S from the baseline. Similarly, for the end of the rise time / beginning of the fall time a similar offset is used. In this case the threshold value c_P (P for peak) is defined as

$$c_P = \min(S) + 0.025 \cdot |B - \min(S)|\,,$$

so 2.5 % of the amplitude above the minimum of the signal S. The position where either threshold is crossed in registers is based on where the simple moving average (of window size 5) crosses c_B and c_P. The number of registers between c_B and c_P defines the rise time (left of the peak) and fall time (right of the peak). [14] At the used clock frequency of 1 GHz each register corresponds to 1 ns in time.

A reconstructed FADC spectrum including indications for baseline, rise and fall time as well as the minimum is shown in fig. 8.3 [15], together with the corresponding event on the Septemboard.

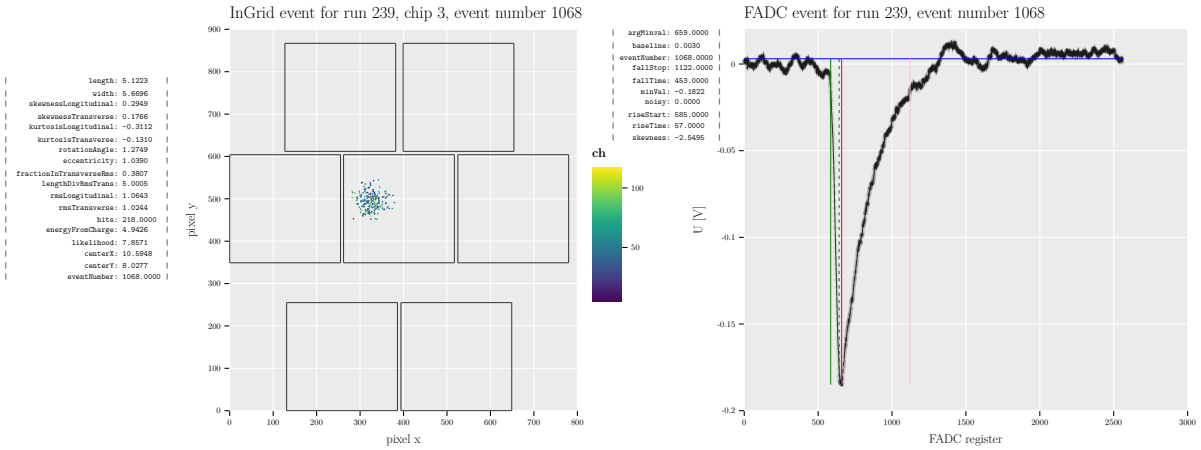


Figure 8.3.: Example of a fully reconstructed InGrid event and FADC spectrum from a 5.9 keV X-ray recorded with the Septemboard detector during a calibration run. On the left of each plot are all properties computed for the data. In the FADC plot the blue line indicates the baseline. Green vertical: rise time from full to dashed line. Red vertical: point of spectrum minimum. Light red: fall time from dashed to full line. Rise / fall time stops 2.5 % before baseline is reached.

[14] The naming of the rise and fall time in the context of a negative pulse is slightly confusing. Rise time refers to the negative rise towards the minimum of the pulse and the fall time to the time to return-to-baseline.

[15] Excuse the small text for the annotations. They are not important, but may be interesting for some readers!

8.6. Scintillator data

For the scintillator signals we only record a trigger flag and the number of clock cycles since the last scintillator trigger from the moment the FADC triggered. These two pieces of information are part of the Septemboard data files included in the header. [16] , [17]

[16] Important note for people potentially investigating the raw data from 2017: There was a small bug in the readout software during the beginning of the 2017 data taking period, which wrote the scintillator trigger clock cycle values into subsequent output files even if no FADC trigger was received (and thus no scintillator trigger was actually read out). However, there is a flag for an FADC trigger. To correctly read the first data runs it is therefore required to not only look at the scintillator trigger clock cycles, but also at whether the FADC actually triggered. This is handled in the analysis framework.

[17] In addition to the above bug, there was unfortunately a more serious bug, which rendered the scintillator counts useless in the end of 2017 / beginning of 2018 data taking period. The polarity of the signals was inverted in the detector firmware, resulting in useless ”trigger” information.

9. Detector installation & data taking at CAST CAST

Contents

In this chapter we will cover the data taking with the Septemboard detector at the CAST experiment. We will begin with a timeline of the important events and the different data taking periods to give some reference and put certain things into perspective, sec. 9.1. We continue with the detector alignment in sec. 9.2, as this is important for the position uncertainty in the limit calculation. Then we discuss the detector setup behind the LLNL telescope, sec. 9.3. Two sections follow focusing on where things did not go according to our plans, a window accident in sec. 9.4 and general issues encountered in each run period in sec. 9.5. We conclude with an overview of the of the total data taken at CAST, sec. 9.6.

For an overview of the technical aspects of the CAST setup and operation see the appendix D. It contains details about the operating procedures with respect to the gas supply and vacuum system, interlocks and more. As the details of that are not particularly relevant after shutdown of the experiment, it is not discussed here.

9.1. Timeline

The Septemboard detector was prepared for data taking at the CAST experiment in July 2017 for preliminary alignment and fit tests. The detector beamline was prepared behind the LLNL telescope and aligned with a laser from the opposite side of the magnet using an acrylic glass target on *<2017-07-07 Fri>* (see fig. 9.1(a)). Vacuum leak tests were performed and the detector installed on *<2017-07-10 Mon>* (see fig. 9.1(b)). In addition, geometer measurements were done for final alignment and as a reference measurement the day after. An Amptek COOL-X X-ray generator [1] ('X-ray finger') was installed on the opposite side of the magnet. A calibration measurement with the X-ray finger ran from *<2017-07-13 Thu>* over night. The aim of an X-ray finger run is to roughly verify the focal spot of the X-ray telescope. After this initial test the detector was dismounted to make space for the KWISP experiment.

Two months later the detector was remounted between *<2017-09-11 Mon>* to *<2017-09-14 Thu>* with another geometer measurement on the last day. During an attempt to clean the detector water cooling system on *<2017-09-19 Tue>*, the window of the detector was destroyed (see section 9.4). This required a detector dismount and transport to Bonn for repairs as the detector was electronically dead after the incident.

Near the end of October *<2017-10-23 Mon>* the remount of the detector started and was finished by *<2017-10-26 Thu>* in time for another geometer measurement and alignment. The next day the veto paddle scintillator was calibrated using a 3-way coincidence in the RD51 laboratory (see sec. 7.3), followed by the installation of the lead shielding and scintillator installation another day later. With everything ready, data taking of the first data taking period with the Septemboard detector started on *<2017-10-30 Mon>*. During the period until *<2017-12-22 Fri>* few minor issues were encountered, see sec. 9.5.1. As CERN is typically closed over Christmas and well into January, data taking was paused until *<2017-02-17 Fri>* (further time is necessary to prepare the magnet for data taking again).

The second part of the first data taking then continued on until *<2017-04-17 Mon>*, with a few more small problems encountered, see 9.5.2. After data taking concluded, dismounting of the detector began the next day by removing the veto scintillator and the lead shielding. On *<2018-04-20 Fri>* another X-ray finger run was performed to get a sense of the placement of the detector during its actual mount as it was during the first data taking period. Afterwards, the detector was fully removed by *<2018-04-26 Thu>* to bring it back to Bonn to fix a few problems.

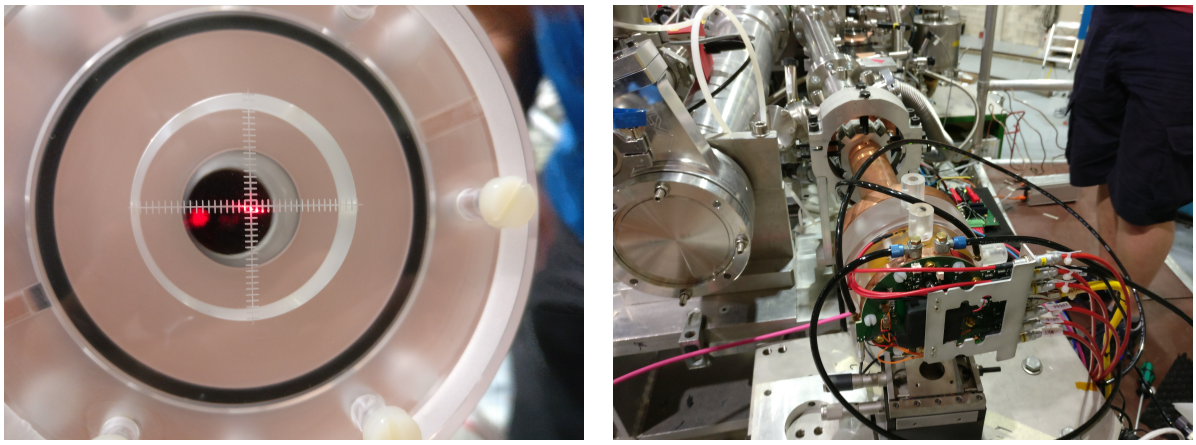
Data taking was initially intended to continue by summer of 2018. The fully repaired detector was installed between *<2018-07-16 Mon>* and *<2018-07-19 Thu>* with a few minor delays due to a change in mounting of the lead shielding support to accommodate a parallel data taking with

[1] https://www.amptek.com/internal-products/obsolete-products/cool-x-pyroelectric-x-ray-generator

KWISP. For alignment another geometer measurement was performed on *<2018-07-23 Mon>*. Unfortunately, external delays pushed the begin of the data taking campaign back into late October. On *<2018-10-20 Sat>* the data taking finally begins after a power supply issue was fixed the day before. The issues encountered during this data taking period, which lasted until *<2018-12-20 Thu>* are mentioned in sec. 9.5.3.

With the end of 2018 the data taking campaign of the Septemboard was at an end. The detector was moved over from CAST to the CAST Detector Lab (CDL) on *<2019-02-14 Thu>* for a measurement campaign behind an X-ray tube for calibration purposes. Data was taken until *<2019-02-21 Thu>* with a variety of targets and filters (covered in sec. 11.2 later). Afterwards the detector was dismounted and taken back to Bonn.

For the results of the different alignments, further see section 9.2.



(a) Laser alignment

(b) Detector installed after alignment

Figure 9.1.: (a) Alignment of the telescope side pipes using an acrylic glass flange with a centered grid and a laser aligned to the magnet bore. The central laser spot is the point on the vertical line extending out from the center. The other points towards the lower right are further refractions. This was better visible by eye. (b) Detector installed on the beamline behind the LLNL telescope on <2017-07-10>.

9.2. Alignment

Detector alignment with the X-ray telescope, the magnet and by extension the solar core during solar tracking is obviously crucial for a helioscope for a good physics result. The alignment procedure used for the Septemboard detector is a three-fold approach:

1. alignment of the piping up to the detector using an acrylic glass target with a millimeter spaced cross, as seen in fig. 9.1(a). This target is mounted to the vacuum pipes in the same way the detector is mounted. A laser is installed on the opposite side of the magnet. With the magnet

bores fully open the laser is aligned such that it propagates the full bore and is reflected by the X-ray telescope into the focal spot. This alignment guarantees the focal spot location to be near the center of the detector. Uncertainty is introduced due to the need to remove the acrylic glass target and install the detector, as the mounting screws allow for small movements. In addition the vacuum pipes are also not perfectly fixed.

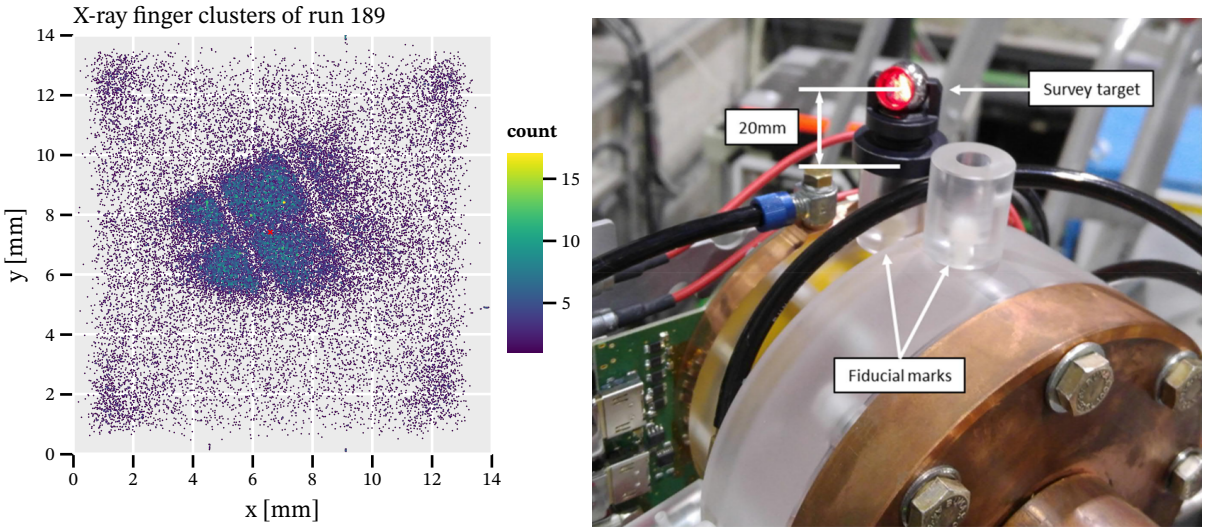
2. alignment of the fully installed detector using an X-ray finger. The 'X-ray finger' is a small electric X-ray generator (in particular an Amptek COOL-X), which is installed in the magnet bore at the opposite end of the magnet. The generated X-rays must traverse the magnet and telescope, thereby being focused by the telescope into the focal spot. As the X-ray finger does not emit parallel light, the resulting distribution of the X-rays on the detector is not a perfect focal spot, even if the telescope was perfect and the detector placed right in the focus. The close distance also implies the focal length is slightly different than for an infinite source. The mean position of the taken data can anyhow be used to determine the likely focal spot position. See below, fig. 9.2(a), for an example and the resulting position from one of the X-ray finger runs.

3. alignment by the geometer group at CERN. A theodolite is installed in the CAST hall and the location of many targets on the magnet, telescope, vacuum pipes and the detector itself are measured up to 0.5 mm precision at 1σ level. See fig. 9.2(b) for a picture of such a target. The initial geometer measurement from *<2017-07-11 Tue>* mainly serves as a baseline reference. As the first two alignment procedures provide a good alignment, a measurement of the existing position by the geometers can then later be used to re-align the detector after it was removed relative to the previous baseline position relative to the telescope. This assures the detector can be remounted and placed in the right location without the need for an additional laser alignment.

The X-ray finger run taken in April 2018 can be used as a reference for the alignment as used during the first data taking. The center positions of each cluster can be shown as a heatmap, where the number of hits each pixel received is colored. Computing the mean position of all those clusters yields the most likely center position of the focal spot. See fig. 9.2(a) for an example of this. The position of the center based on the mean of all cluster centers is about 0.4 mm away from the center in both axes.

With this setup after each remounting a geometer measurement was performed to align the detector back to the initial laser alignment. As the second mounting of the detector in September 2017 was not used for any data taking, the associated geometer measurement is irrelevant.

Tab. 9.1 summarizes the values of the geometer alignment results for each of the measurements using the CenterR and CenterF positions defined based on the initial geometer measurement in July



(a) X-ray finger clusters

(b) Geometer target

Figure 9.2.: (a): Cluster center positions of the X-ray finger run 189 from April 2018. The red cross marks the center of all cluster centers, which is the most likely position of the focal spot. It is $\sim$ 0.4 mm away from the chip center in both axes. The two parallel lines with less clusters are the window strongback. The orthogonal line is a graphite spacer in the center of the LLNL telescope. (b): Image showing the targets on the detector. The acrylic glass cylinders are the fiducial marks used to hold the actual survey target. The survey target is a mirror to reflect the laser of the theodolite. Image from [15].

2017. In each case the shifts in X, Y and Z direction is usually significantly less than 1 mm.

For a detailed overview of the geometer measurements see the public EDMS links under [15, 16, 17, 14, 13] containing the PDF reports for each measurement.

9.3. Detector setup at CAST

The setup of the full beamline from the magnet end cap to the detector is shown in a render in fig. 9.3. The piping shows a clear kink introduced using a flexible bellow. This setup is used to move the detector mount further away from the other beamline to provide more space for two setups side-by-side. At the same time it is an artifact of the LLNL telescope only being a 30 ° portion of a full telescope resulting in the focal plane not being centered in front of the telescope. Not shown in the image is the lead shielding installed around the detector as well as the veto scintillator, which covers the majority of the beamline area. The lead shielding is a 5 to 15 cm thick 'castle' of lead around the detector (10 cm on top and behind, 15 cm in front and 5 cm and 10 cm on each side). An annotated image of the real setup is seen in fig. 9.4, which shows lead shielding, veto scintillator, ^{55}Fe source manipulator and the LLNL X-ray telescope. The setup is behind the VT3 gate valve of the CAST magnet.

Measurement	Target	ΔX [mm]	ΔY [mm]	ΔZ [mm]	Useful
11.07.2017					yes
14.09.2017	CenterR	-0.1	0.3	-0.8	no
	CenterF	-0.1	0.3	-0.9	
26.10.2017	CenterR	0.2	0.6	0.2	yes
	CenterF	0.1	0.6	-0.1	
24.04.2018	CenterR	0.5	0.5	0.0	yes
	CenterF	0.4	0.5	-0.3	
23.07.2018	CenterR	1.1	0.5	0.6	yes
	CenterF	1.0	0.5	0.3	

Table 9.1.: Overview of the results of the different geometer alignment measurements. The first measuremnt serves as the baseline to define 2 points (CenterR and CenterF) relative to which alignment later is done. The initial alignment is done both by laser and X-ray finger. The second geometer measurement is not useful, as no data was taken with it, due to the window rupture accident.

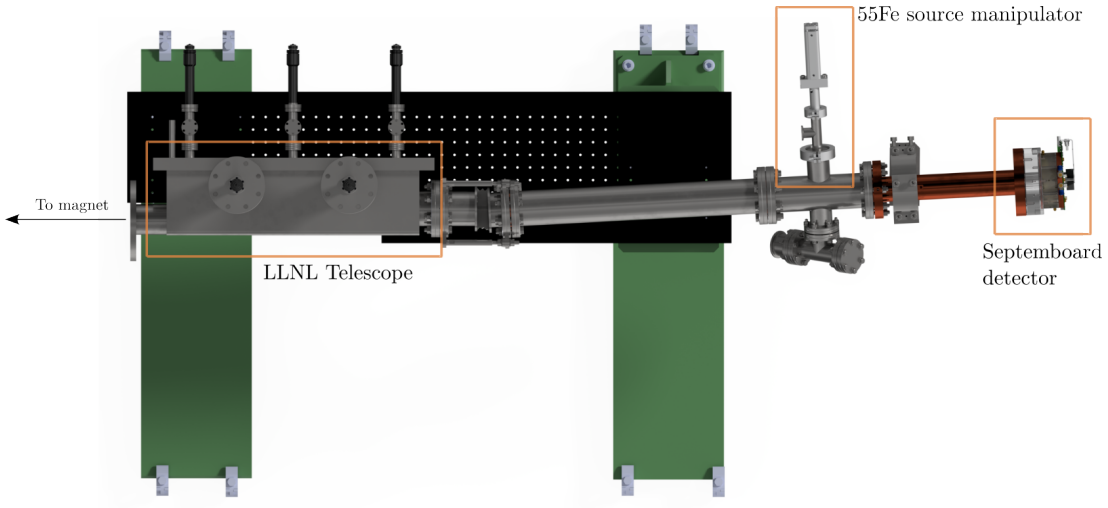


Figure 9.3.: Render of the detector setup up to the magnet end cap as seen from above. The beamline kinks away from the other beamline (”below” in this image) to provide more space for two detectors at the same time. Image courtesy of Tobias Schiffer.

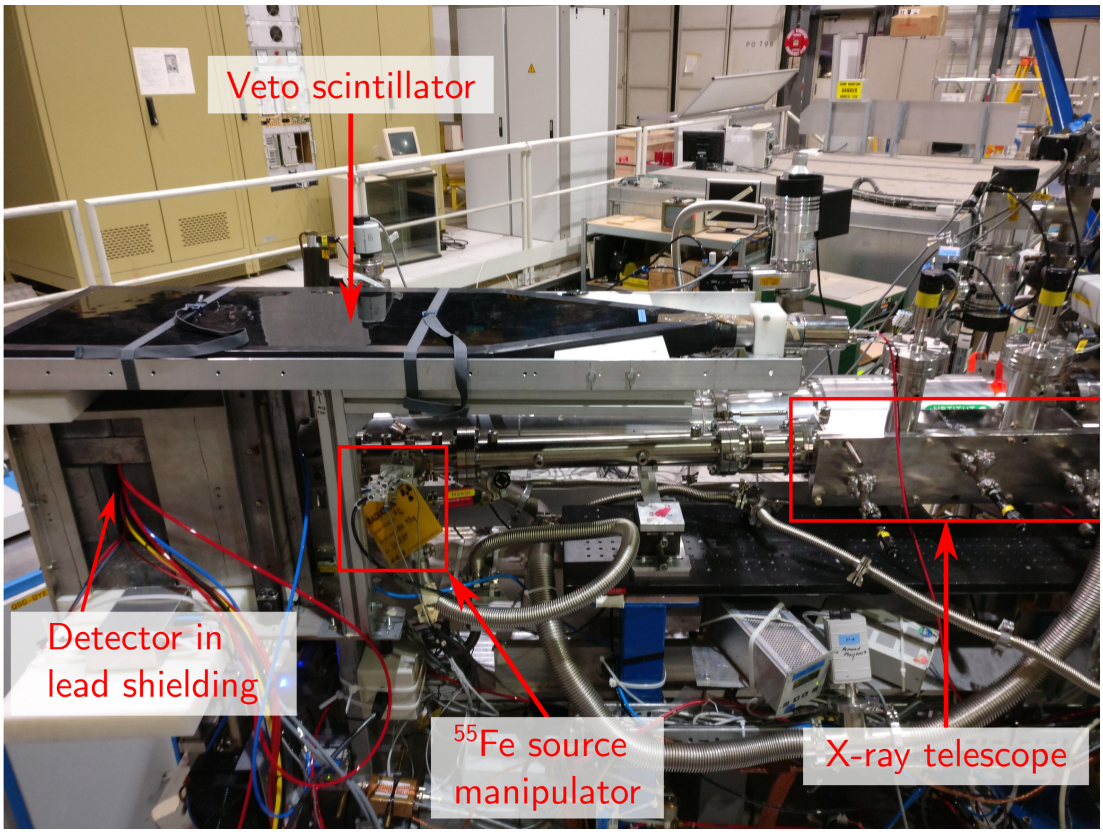


Figure 9.4.: Annotated setup as installed in October 2017 for the first data taking campaign. The detector is seen in its lead shielding, with the veto scintillator covering a large angular portion above the detector. The ^{55}Fe source manipulator is seen head-on here. On the right towards the magnet we see the housing of the LLNL X-ray telescope.

9.3.1. ^{55}Fe source and manipulator

As seen in the previous section the setup includes a ^{55}Fe source. Its purpose is both monitoring of the detector behavior and it serves as a way to calibrate the energy of events (as mentioned in theory section 5.3.8). More details on the usage and importance for data analysis will be given in chapter 10. It is installed on a pneumatic manipulator. Using a compressed air line with about 6 bar pressure the manipulator can be moved up and down. Under vacuum conditions of the setup the manipulator is inserted unless the compressed air is used to push it out.

A Raspberry Pi [2] is installed close to the manipulator and connects to the two Festo [3] control sensors at the top and bottom end of the manipulator using the general purpose input/output (GPIO) pins. Two pins are used to read the sensor status from each. Five more pins connect to a 24 V relay, which is used to control the controllers for the compressed air line. The relay is controlled by pulse width modulation (PWM). The software controlling the GPIO pins of the Raspberry Pi is written in Python. A client program is running on a computer in the CAST control room and communicates with the Raspberry Pi using a network connection on which a server process is running. It can receive connections through a socket, allowing for remote and programmatic control of the manipulator via a set of simple string based messages. Further, it provides a REPL (read-evaluate-print loop) to control it interactively. For more details about the software see the extended version of this thesis.

9.4. Window accident

During the preparations of the detector for data taking, it became clear that the rubber seals of the quick connectors used for the water cooling system started to disintegrate. The connectors were replaced by Swagelok connectors, but the water cooling system still contained rubber pieces blocking the flow. Due to the small diameter and twisted layout of the cooling ducts in the copper body, the only way at hand to clean them was a compressed air line, normally used for operation of the ^{55}Fe manipulator (see sec. 9.3.1). This cleaning process worked very well. Multiple cleaning & water pumping cycles were needed, as after cleaning the system with compressed air, pumping water the next time moved some remaining pieces, which blocked it again. After multiple cycles at which no more clogging happened upon water pumping a final cycle was intended. As the gas supply and the water cooling system after replacement of the quick connectors now used not only the same tubing, but also the same connectors, the compressed air line was mistakenly connected to the gas supply instead of water cooling line by me. The windows – tested up to 1.5 bar pressure – could not withstand the sudden pressure of the compressed air line of about 6 bar. A sudden and catastrophic window

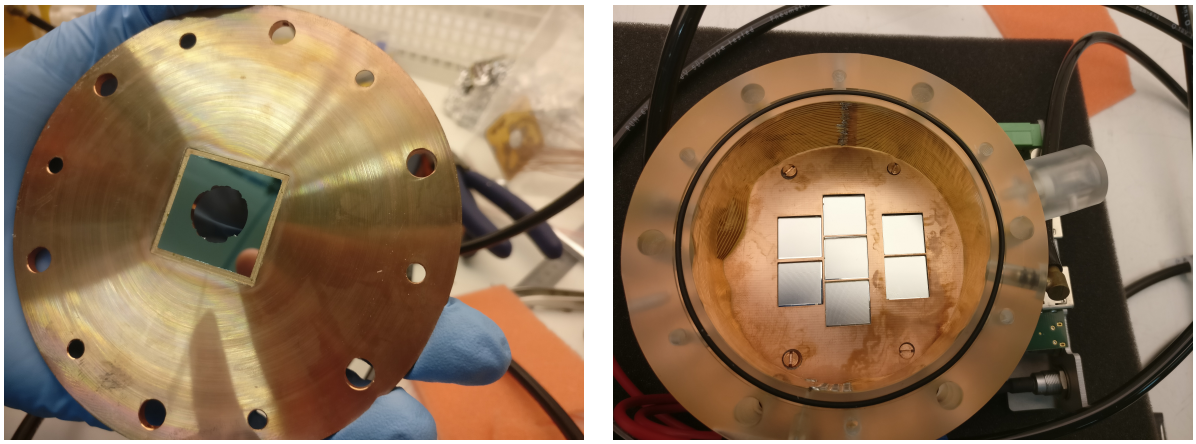
[2] https://www.raspberrypi.org/about/
[3] https://www.festo.com/

failure broke the vacuum and shot window pieces as well as possible contamination into the vacuum pipes towards the X-ray optics.

Because the LLNL telescope is an experimental optics there was worry about potential oil contamination coming from dirty air of the compressed air line. A conservative estimate of this given an upper bound on contamination of the air, volume of the vacuum pipes and the telescope area was computed. Assuming a flow of compressed air of 5 s, a ISO 8573-1:2010 class 4 compressed air contamination of $\text{ppmv}_{\text{oil}} = 10\,\text{mg}\,\text{m}^{-3}$ and all oil in the air sticking to the telescope shells would lead to a contamination of $c_{\text{oil}} = 41.7\,\text{ng}\,\text{cm}^{-2}$. More realistic is about 1 % of that due to the telescope only being less than $\frac{1}{10}$ of the full system area and the primary membrane pump likely removing the majority ($> 90\,\%$) of the oil in the first place. This puts an upper limit of $c_{\text{oil}} = 0.417\,\text{ng}\,\text{cm}^{-2}$, which is well below anything considered problematic for further data taking.

Further, the ^{55}Fe source manipulator likely caught most of the debris, as it was fully inserted due to the necessary removal of the compressed air line from it, which is normally needed to keep the manipulator extruded when the system is under vacuum. For this reason it is unlikely any window debris could have caused significant scratches in the telescope layers.

After the incident the detector was dismounted and taken to the CAST detector lab. Fig. 9.5(a) shows the detector from above with the small remaining pieces of the window. Fig. 9.5(b) shows the detector inside after opening it. A bulge is visible where the gas inlet is and the compressed air entered. As the detector was electronically dead after the incident, the decision was made to move it back to Bonn for repairs. It turned out that the Septemboard had become loose from the connector.



(a) Broken window from the inside

(b) View into the detector after accident

Figure 9.5.: (a) shows the cathode of the detector from the inside with the broken window. Essentially the full window directly exposed to vacuum is gone. (b) is the view into the detector without the cathode. A bulge of the field cage is visible where the compressed air entered.

9.5. Data taking woes

In this section we will cover the smaller issues encountered during the data taking. These are worth naming, due to having an impact on the quality of the data as well as affecting certain aspects of data analysis. In case someone wishes to analyze the data, they should be aware of them. We will cover each of the effectively three data taking periods one after another.

9.5.1. 2017 Oct - Dec

The first data taking period from *<2017-10-30 Mon>* to *<2017-12-22 Fri>* initially had a bug in the data acquisition software, which failed to reset the veto scintillator values from one event to the next, if the next one did not have an FADC trigger. In that case in principle the veto scintillators should not have any values other than `0`. However, as there is a flag in the data readout for whether the FADC triggered at all, this is nowadays handled neatly in the software by only checking the triggers if there was an FADC trigger in the first place. Unfortunately, it was later found that the scintillator triggers were nonsensical in this data taking period due to a firmware bug anyway.

Starting from the solar tracking run on *<2017-11-29 Wed>* the analogue FADC signals showed significant signs of noise activity. This lead to an effectively extremely high dead time of the detector, because the FADC triggered pretty much immediately after the Timepix shutter was opened. As I was on shift during this tracking, I changed the FADC settings to a value, which got rid of the noise enough to continue normal data taking. The following changes were made:

- differentiation time reduced from 50 ns to 20 ns
- coarse gain of the main amplifier increased from `6x` to `10x`

Evidently this has a direct effect on the shape of the FADC signals, to be discussed in sec. 10.4.1.

On *<2017-12-05 Tue>* while trying to investigate the noise problem which resurfaced the day before despite the different settings, a fuse blew in the gas interlock box. This caused a loss of a solar tracking the next day. The still present FADC noise lead me to change the amplification settings more drastically on *<2017-12-07 Thu 08:00>* during the shift:

- integration time from 50 ns to 100 ns

The same day in the evening the magnet quenched causing the shift to be missed the next day. In the evening of *<2017-12-08 Fri>* the integration time was turned down to 50 ns again, as the noise issue was gone again.

A week later the integration time was finally changed again to 100 ns. By this time it was clear that there would be no easy fix to the problem and that it is strongly correlated to the magnet activity during a shift. For that reason the setting was kept for the remaining data taking periods.

9.5.2. 2018 Feb - Apr

Two days before the data taking period was supposed to start again in 2018, there were issues with the detector behavior with respect to the thresholds and the gain of the GridPixes. During one calibration run with the ^{55}Fe source the effective gain dropped further and further such that instead of $\sim$ 220 electrons less than $\sim$ 100 were recorded. This turned out to be a grounding issue of the detector relative to the water cooling system.

Further, the temperature readout of the detector did not work anymore. It is unclear what happened exactly, but the female micro USB connector on the detector had a bad soldering joint as was found out after the data taking campaign. It is possible that replugging cables to fix the above mentioned issue caused an already weak connector to fully break.

The second data taking period finally started on *<2018-02-17 Sat>* and ran until *<2018-04-17 Tue>*.

This data taking campaign still ran without functioning scintillators, due to lack of time and alternative hardware in Bonn to debug the underlying issue and develop a solution.

9.5.3. 2018 Oct - Dec

Between the spring and final data taking campaign the temperature readout as well as the firmware were fixed to get the scintillator triggers working correctly, with the installation being done end of July 2018. By the time of the start of the actual solar tracking data taking campaign at the end of October however, a powering issue had appeared. This time the Phoenix connector on the intermediate board had a bad soldering joint, which was finally fixed *<2018-10-19 Fri>*. Data taking started the day after.

Two runs in mid December showed strong noise on the FADC again. This time no amount of changing amplifier settings had any effect, which is why 2 runs were done without the FADC. See runs 298 and 299 in the appendix, tab. E. For the last runs it was activated again and no more noise issues appeared.

9.5.4. Concluding thoughts about issues

The FADC noise issue was in many ways the most disrupting active issue the detector was plagued by. In hindsight, the standard LEMO cable used should have been a properly shielded cable. Someone with more knowledge about RF interference should have assisted in the installation. In a later section, 10.4.1, the typical signals recorded by the FADC under noise will be shown as well as mitigation strategies on the software side. Also how the signals and the FADC activation threshold changed due to the changed settings will be presented.

9.6. Summary of CAST data taking

In summary then, the data taken at CAST with the Septemboard detector can be split into two periods. The first from October 2017 to April 2018 and the second from October 2018 to December 2018. The former will from here on be called "Run-2" and the latter "Run-3". Run-1 refers to the data taking campaign with the single GridPix detector in 2014 and 2015. The distinction of run periods is mainly based on the fact that the detector was dismounted between Run-2 and Run-3 and additionally a full detector recalibration was performed, meaning the datasets require slightly different parameters for calibration related aspects.

During Run-2 the scintillator vetoes were not working correctly. The FADC was partially noisy. In Run-3 all detector features were working as intended. The feature list is summarized in tab. 9.2.

Feature	Run 2	Run 3
Septemboard	o	o
FADC	m	o
Veto scinti	x	o
SiPM	x	o

Table 9.2.: Overview of working (o), mostly working (m), not working (x) features in each run. FADC was partially noisy in Run-2.

Run-2 ran with a Timepix shutter time of `2/32` (ref. sec. 8.4.4) resulting in about 2.4 s long frames. This was changed with the start of 2018 (still in Run-2) to `2/30` ($\sim$ 2.2 s).

In total 115 solar trackings were recorded between Run-2 and Run-3, out of 120 solar trackings taking place. 4 of the 120 total were missed for detector related reasons and one was aborted after 30 minutes of tracking time. This amounts to about 180 h of tracking data. Further, 3526 h of background data and 194 h of ^{55}Fe calibration data were recorded. The total active fraction of these times is about 90 % in both run periods. See tab. 9.3 for the precise times and fractions of active data taking. Two X-ray finger runs were done for alignment purposes (out of which only 1 is directly useful).

Outside the issues mentioned in the previous section 9.5, the detector generally ran very stable. Certain detector behaviors will be discussed later, which do not affect data quality as they can be calibrated out.

Table 9.4 provides a comprehensive overview of different statistics of each data taking period, split by calibration and background / solar tracking data. The appendix E lists the full run list with additional information about each run. Further, appendix J shows occupancy maps of the Septemboard for Run-2 and Run-3, showing a mostly homogeneous activity, as one would expect for background data taking. Fig. L.1 in appendix L shows the raw rate of activity on the center chip over the entire

	Solar tracking [h]	Active s. [h]	Background [h]	Active b. [h]	Active [%]
Run-2	106.006	93.3689	2391.16	2144.12	89.65
Run-3	74.2981	67.0066	1124.93	1012.68	90.02
Total	180.3041	160.3755	3516.09	3157.35	89.52

Table 9.3.: Overview of the total data taken with the Septemboard detector at CAST in the time between October 2017 and December 2018. 'Active s' and 'Active b.' refers to the total solar tracking and background time excluding the dead time due to readout of the septemboard. See the table below 9.4 for more details.

CAST data taking period.

Field	calib Run-2	calib Run-3	back Run-2	back Run-3
total duration	107.42 h	87.06 h	2497.16 h	1199.22 h
active duration	2.6 h	3.53 h	2238.78 h	1079.6 h
active fraction	2.422 %	4.049 %	89.65 %	90.02 %
# trackings	0	0	68	47
non tracking time	107.42 h	87.06 h	2391.15 h	1124.93 h
active non tracking time	2.6 h	3.53 h	2144.11 h	1012.67 h
tracking time	0 h	0 h	106.01 h	74.3 h
active tracking time	0 h	0 h	93.36 h	67 h
Events				
total # events	532 020	415 927	3 758 960	1 837 330
only center chip	472 048	361 244	21 684	10 342
only any outer chip	5	5	1 558 546	744 722
center + outer	59 554	53 499	1 014 651	486 478
center chip	531 602	414 743	1 036 335	496 820
any chip	531 607	414 748	2 594 881	1 241 542
fraction with center	99.92 %	99.72 %	27.57 %	27.04 %
fraction with any	99.92 %	99.72 %	69.03 %	67.57 %
with fadc readouts	531 529	413 853	542 233	211 683
fraction with FADC	99.91 %	99.50 %	14.43 %	11.52 %
with SiPM trigger <4095	1656	20	8585	4304
with veto scinti trigger <4095	0	2888	0	70 016
with any SiPM trigger	531 528	1312	825 460	34 969
with any veto scinti trigger	0	216 170	0	206 025
fraction with any SiPM	99.91 %	0.3154 %	21.96 %	1.903 %
fraction with any veto scinti	0.000 %	51.97 %	0.000 %	11.21 %

Table 9.4.: Overview of the data taken in each of the runs split by calibration data ("calib") and background ("back"). First information about the total recorded time and trackings and then event information regarding general activity and activity split by chips, FADC and scintillators. Note that the scintillator information for Run-2 is not useful, as the signals recorded were not actually real signals.

10. Data calibration Calibration

Contents

With the roughly 3500 h of data recorded at CAST it is time to discuss the final calibrations [1] necessary for the calculation of a physics result. On the side of the Septemboard detector this means the 'energy calibration', sec. 10.1; the calculation of the energy of each event recorded with the Septemboard detector. This necessarily needs to include a discussion of detector variability both due to external factors as well as differing detector calibrations and setups, sec. 10.2. Here we provide the reasoning for the choices leading to the final energy calibration, presented in sec. 10.3. Similarly, for the FADC the impact of the noise seen during data taking and resulting differing amplifier settings will be discussed in sec. 10.4.

10.1. Energy calibration - in principle

The reconstructed data from the GridPixes, as described in chapter 8.4 (cluster finding, cluster reconstruction and charge calibration), still needs to be calibrated in energy. The charge calibration 7.1.1 computes the number of electrons recorded on each GridPix pixel in an event from the ToT counts.

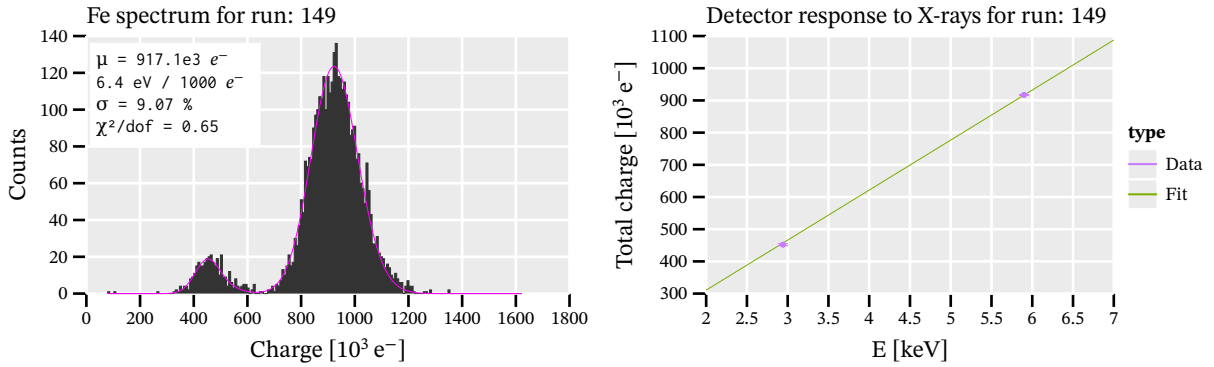
In order to calculate an equivalent energy based on a certain amount of charge – which depends on the gas gain – the data recorded using the ^{55}Fe calibration source at CAST is used. As the ^{55}Fe spectrum (see sec. 5.3.8) has a photopeak at 5.9 keV and an escape peak at 2.9 keV it provides two different

[1] Note that the term 'calibration' is a heavily loaded term implying very different things depending on context. This can at times be confusing. I try to be explicit by fully specifying what calibration is meant when it might be ambiguous.

lines relating charges to energies for calibration. While the charge calibration for each pixel from ToT to electrons is non-linear, the relation between energy and recorded charge is linear. The position of the two peaks in the ^{55}Fe spectrum needs to be determined precisely, which is done using a double gaussian fit

$$f(N_e, \mu_e, \sigma_e, N_p, \mu_p, \sigma_p) = G^{\text{esc}}_{\text{K}_\alpha}(N_e, \mu_e, \sigma_e) + G_{\text{K}_\alpha}(N_p, \mu_p, \sigma_p), \tag{10.1}$$

where G is a regular gaussian, one for the escape peak G^{esc} and one for the photopeak G. An example spectrum with such a fit can be seen in fig. 10.1(a).



(a) ^{55}Fe spectrum (b) Energy calibration

Figure 10.1.: (a): Fit to a ^{55}Fe calibration run from the CAST data (run 149) using a double gaussian fit. (b): Linear fit to the escape and photopeak energies to relate charges in electrons to energies in keV.

Then, a linear function without y-offset

$$Q(E) = m_c \cdot E$$

is fitted to the found peak positions of the spectra by charge Q, against the known energies E of the peaks in the ^{55}Fe spectrum. This yields the calibration factor, $a = m_c^{-1}$, which can be used to calibrate all events with the same gas gain. Over the time of data taking at CAST the gas gain varies by a significant margin, requiring a more complex calibration routine as the calibration factor would produce too imprecise energy values otherwise (for example if each ^{55}Fe calibration run were used to deduce one calibration factor $a = m_c^{-1}$ to be applied to the closest background data in time). An example for this fit is seen in fig. 10.1(b).

Fortunately, the gas gain can be computed using raw data without evaluating any physical events, allowing calculation of it also for raw background data. This motivates the idea to map a gas gain to a calibration factor needed to calibrate events at such gas gains in energy. Taking a certain time interval in which the detector gas gain is assumed constant, the gas gain of all time slices of this length is computed for background and calibration data. For all time slices in the calibration data

the procedure above – fitting the ^{55}Fe spectrum and calculating the energy calibration – is performed. A higher gas gain leads to linearly more recorded electrons in the ^{55}Fe spectra. Therefore, all energy calibration factors determined from different time intervals should be on a slope depending on the gas gain. As such a final fit

$$a(G) = m_g \cdot G + b \tag{10.2}$$

is performed to all time intervals of all calibration runs. This yields the energy calibration factor a valid for a given gas gain G. Then in order to calibrate the energy of a given cluster in the background data, the same time slicing is performed and one gas gain calculated for each slice. The gas gain is inserted into the fit and the resulting calibration factor is used to calibrate the energy of every cluster in that time slice. We will come back to this fit in sec. 10.3 to finalize the energy calibration.

The remaining question is the stability of the gas gain over time, which we will look at next in the context of the general detector behavior over time. This allows us to find a suitable time interval to use for all data and hence perform a temporally stable energy calibration.

10.2. Detector behavior over time

Outside the detector related issues discussed in section 9.5 the detector generally ran very stable during Run-2 and Run-3 at CAST. This allows and requires to assess the data quality in more nuanced ways. Specifically, the stability of the recorded signals over time is of interest, which is one of the main purposes of the ^{55}Fe calibration runs. A fixed spectrum allows to verify stable operation easily. In particular of interest for the energy calibration of the data are the detected charge and gas gain of the detector.

As the charge and gas gain can be computed purely from individual pixel data without any physical interpretation, it serves as a great reference over time. Longer time scale variations of the gas gain were already evident from the calibration runs during data taking and partially expected due to the power supply and grounding problems encountered, as well as different sets of calibrations between Run-2 and Run-3. By binning the data into short intervals of order one hour, significant fluctuations can be observed even on such time scales. Fig. 10.2 shows the normalized median of the total charge in events for all CAST data normalized by the datasets (background and calibration). Each data point represents a 90 min time slice. Some data is removed prior to calculation of the median as mentioned in the caption. The important takeaway of the figure is the extreme variability of the median charge (up to 30 %!). Fortunately though, the background and calibration data behaves the same, evident by the strong correlation (purple background, green calibration). While the causes for the variability are not entirely certain (see sec. 10.2.1), it allows us to take action and calibrate the data accordingly.

Figure 10.2.: The plot shows the median charge within 90 min time windows of both background and calibration data. Some data is removed (only clusters with less than 500 pixels active to remove the worst sparks and extremely large events) and only events within the inner 4.5 mm radius are considered. Each data type (calibration and background) is normalized to 1 as the median charge is very different in the datasets. The median is used instead of the mean to further remove effect of very rare, but extreme outliers. Each pane of the plot shows a portion of data taking with significant time without data between each.

10.2.1. Potential causes for the variability

One possible cause for the variability seen in the previous section is the electronics of the detector readout. Either a floating ground or unstable power supply can result in the activation thresholds of the pixels moving – as was indeed observed – as mentioned in section 9.5. Lesser versions of the problems discussed in that section could theoretically explain the variations. Specifically, in regards to the ^{55}Fe spectra showing variation, the number of pixels and the amount of charge are directly correlated. The number of pixels is plainly a clamped version of the charge information. If electronics caused threshold variations, it would both change the effective `ToT` value as well as the number of pixels activated in the first place. Fortunately, the center chip also contains the FADC, which allows for an independent measurement of the effective charge generated below the grid and thus another indirect measurement of the gas gain. By comparing how the mean position of the ^{55}Fe spectra behave in the FADC data compared to the GridPix data we can deduce if the GridPix behavior likely is due to real gas gain changes or due to electronics.

Fig. 10.3 shows the (normalized) position of the ^{55}Fe photopeak based on a fit to the pixel, charge and FADC spectrum (the latter based on the amplitudes of the FADC signals). Aside from the variations in the FADC data in the 2017 data (left) due to the changed FADC settings (more on that in sec. 10.4), the 'temporally local' changes in all three datasets are almost perfectly correlated. This implies a *real physical origin* in the observed variation and not an electronic or power supply origin.

A physical change in the gas gain can either be caused by a change in high voltage in the amplification region, a change in gas composition or gas properties (assuming no change in the physical size of the amplification gap, which is reasonable at least within Run-2 and Run-3 as the detector was not touched).

Firstly, the high voltage, while not logged to a file [2], was visually inspected regularly and was always kept at the desired voltages by the Iseg HV module within the operating window. It is a very unlikely source of the variability. [3]

Secondly, there is no reason to believe the gas composition to be at fault as a) the detector is used in an open loop at a constant gas flow and b) it would then if anything show up as a sudden change in detector properties upon a gas bottle change and not a continuous change during operation.

This finally leaves the properties of the gas itself, for which three variables are (partially) known:

[2] Once again, in hindsight writing a log file of the high voltage values would have been valuable, especially as it could have been done straight from TOS. However, similar to what lead to losses of the temperature log files, this was simply not prioritized to implement at the time. The same holds for the gas pressure in the chamber, which should have been logged using the `FlowView` software of the Bronkhorst pressure controller used to control it.

[3] From other Iseg HV modules used in lab course experiments we know that when they are faulty it is very evident. We have never experienced a module that reads correct values, but actually supplies the wrong voltage. In each faulty case the desired target voltage was simply not correctly held and this was visible in the read out voltage.

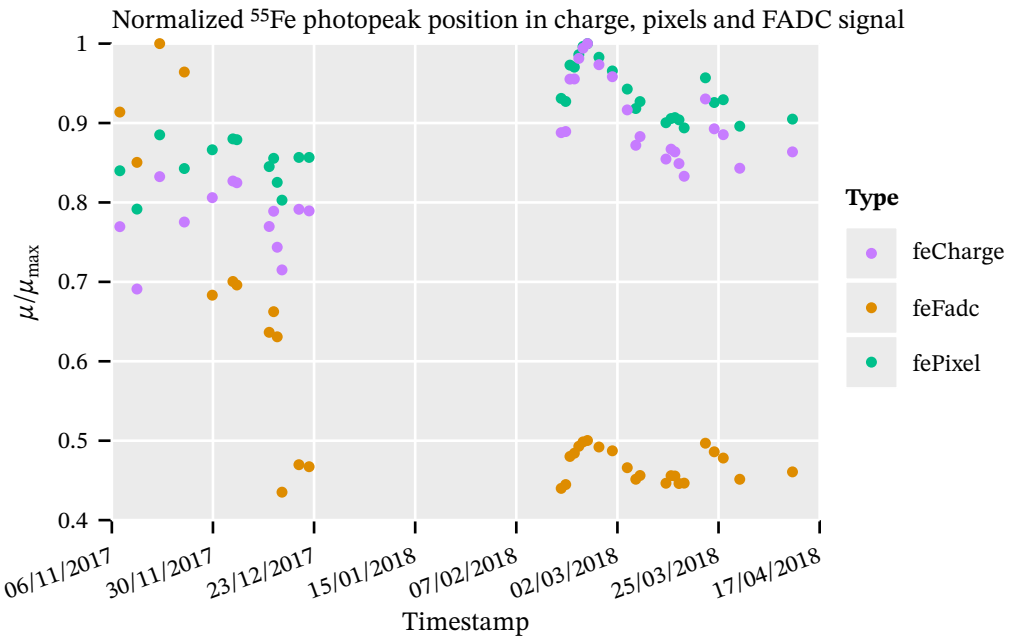


Figure 10.3.: Normalized photopeak positions in the Run-2 data based on the charge (purple), pixel (green) and FADC (orange) spectra. The empty range in the middle is the period between Dec 2017 and Feb 2018. The strong changes in the FADC on the left are due to the different FADC settings. Beyond that the three sets of data are fully correlated, implying a physical origin in the variation. Compare how local (in time) features appear identical in each data.

1. the gas flow
2. the chamber pressure via the pressure controller on the outlet side
3. the temperature

The gas flow was at a relatively constant $2\,\mathrm{L\,h^{-1}}$. The absolute value should not be too relevant, as the flow is small in absolute terms and thus should have no effect on the gas properties in the chamber as such (via gas flow related effects causing turbulence or similar in the chamber). Its secondary impact is only one on absolute gas pressure, which is controlled by the pressure controller, which provides granular control. While also no log files were written for the chamber pressure, visual inspection was also done regularly and the pressure was at a constant 1050 mbar at most varying by 1 mbar in rare cases, but certainly not in a way correlating to the gas gain variations.

This leaves the temperature inside the chamber and in the amplification region as the final plausible source of the variations. As the temperature log files for the Septemboard were lost due to a software bug (more on that in appendix sec. A.2.4), there are two other sources of temperature information. First of all the shift log of each morning shift contains one temperature reading of the Septemboard, which yields one value for every solar tracking. Second of all the CAST slow control log files contain multiple different temperature readings in one second intervals. Most notably the ambient temperature in the CAST hall, which up to an offset (and some variation due to detector load and cooling efficiency) should be equivalent to the gas temperature. Fig. 10.4 shows the normalized temperature sensors in the CAST hall (excluding the exterior temperature) during the Run-3 data taking period together with the normalized peak position of the ^{55}Fe spectra in pixels (black points), the temperature from the shift logs (blue points) and the gas gain values of each chip (smaller points using the color scale, based on 90 min intervals per point). The blue points of the temperature of the Septemboard recorded during each solar tracking nicely follow the temperature trend of the ambient temperature (`T_amb`) in the hall, as expected. Comparing the ^{55}Fe spectra mean positions with the shift log temperatures does not allow to draw meaningful conclusions about possible correlations, due to lack of statistics. But the gas gains of each chip compared to the temperature lines does imply an (imperfect) *inverse* correlation between the temperature and the gas gain.

As discussed in theory sec. 5.3.6 the expectation for the gas gain given constant pressure is $G \propto e^{\alpha}$ where the first Townsend coefficient α scales with temperature by

$$\alpha \propto \frac{1}{T} \exp\left(-\frac{1}{T}\right). \tag{10.3}$$

The combination of the inverse relation to T and its negative exponential is a monotonically increasing sublinear function (and not decreasing as $1/T$ would imply alone) in the relevant parameter ranges. This should imply an increase in gas gain instead of the apparent decrease we see for in-

creasing temperatures. The kind of scaling according to eq. 10.3 was also already experimentally measured for GridPix detectors by L. Scharenberg in [187]. The implications seem to be that the assumptions going into the α scaling must have been violated. The septemboard detector in its – essentially open – gas system is a non-trivial thermodynamic system due to the significant heating of the Timepix ASICs and very small amplification region of 50 µm height enclosing a gas mass, where gas flow is potentially inhibited.

This is not meant as a definitive statement about the origins of the gas gain variations in the Septemboard detector data. *However*, it clearly motivates the need for an even more in depth study of the behavior of these detectors for different gas temperatures at constant pressures (continuing the work of [187]). More precise logging of temperatures and pressures in future detectors is highly encouraged. Further, a significantly improved cooling setup (to more closely approach a region where temperature changes have a smaller relative impact), or theoretically even a temperature controlled setup (to avoid temperature changes in the first place) with known inlet gas temperatures might be useful. This behavior is one of the most problematic from a data analysis point of view and thus it should be taken seriously for future endeavors.

See appendix F for plots similar to fig. 10.4 for the other periods of CAST data taking and a scatter plot of the center chip gas gains against the ambient temperature directly.

10.2.2. Gas gain binning

Motivated by the strong variation seen over timescales much shorter than the typical length of a background run, the gas gain needs to be computed in time slices of a fixed length. This is naturally a trade-off between assigning accurate gas gains to a time slice and acquiring enough statistics to compute said gas gain correctly.

To determine a suitable time window the gas gain was computed for a fixed set of different time intervals and figures similar to fig. 10.2 were considered not only for the median charge, but also different geometric cluster distributions. Further, by applying the energy calibration based on each different set of time intervals to the background data (as will be explained in sec. 10.3), the histograms of the median cluster energy in the background data was studied. The ideal time interval is one in which the resulting median energy distribution has low variance and is unimodal approaching a normal distribution, (background in all slices is equivalent over large enough times) while at the same time provides enough statistics in the ^{55}Fe spectrum of the slice to perform a good fit.

Unimodality can be quantitatively checked using different goodness of fit tests (Anderson-Darling, Cramér-von Mises, Kolmogorov-Smirnov). See appendix F.1 for a comparison and further plots comparing the intervals. The goodness of fit tests tend to favor shorter intervals, in particular 45 min. However, looking at fig. 10.5 shows that the variance grows significantly below 90 min.

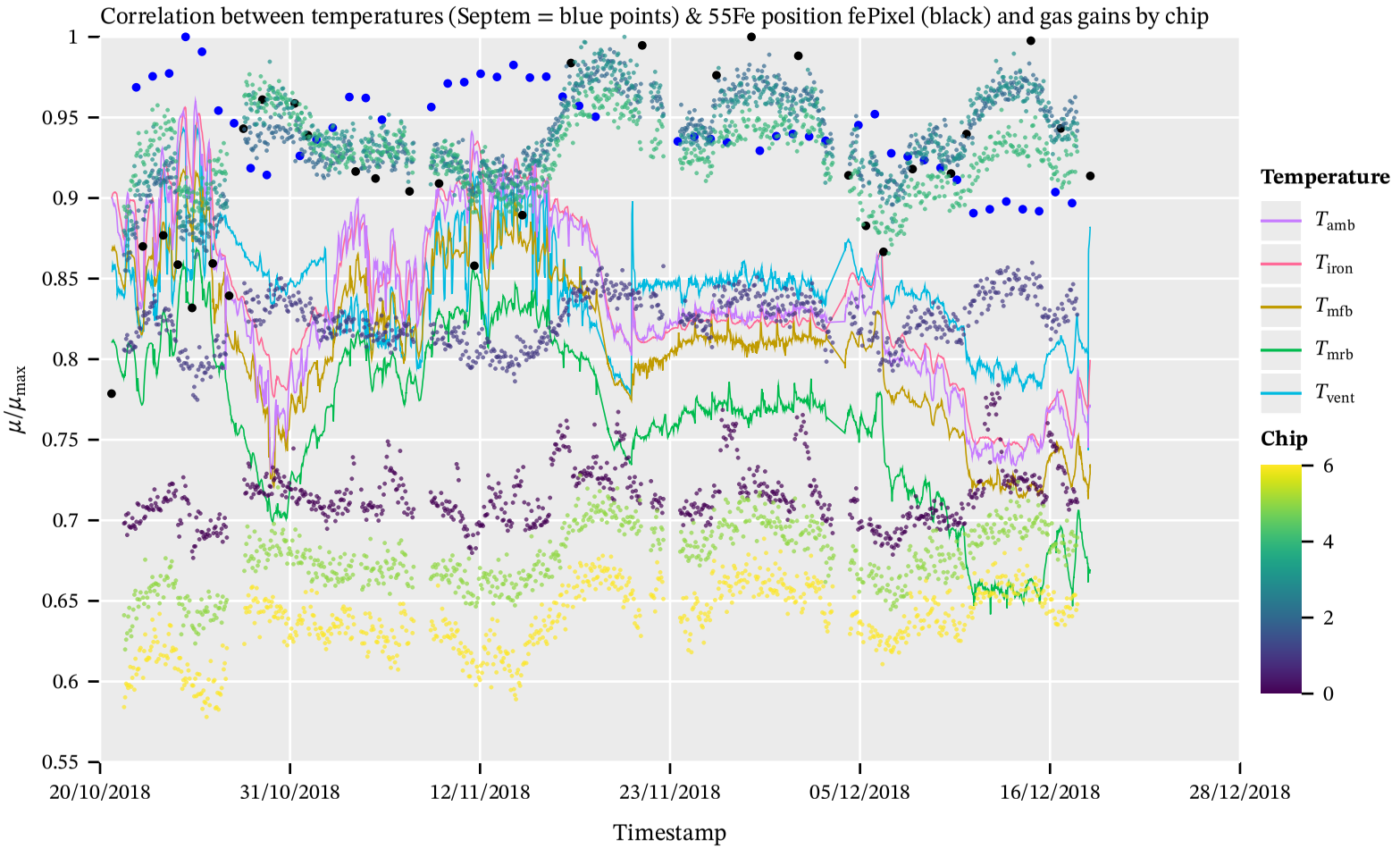


Figure 10.4.: Normalized data for Run-3 of the temperature sensors from the CAST slow control log files compared to the behavior of the mean peak position in the ^{55}Fe pixel spectra (black points), the recovered temperature values recorded during each solar tracking (blue points) and the gas gain values computed based on 90 min of data for each chip (smaller points using Viridis color scale). The shift log temperatures nicely follow the trend of the general temperatures. Gas gains and temperatures seem to be inversely correlated, providing a possible explanation for the detector behavior.

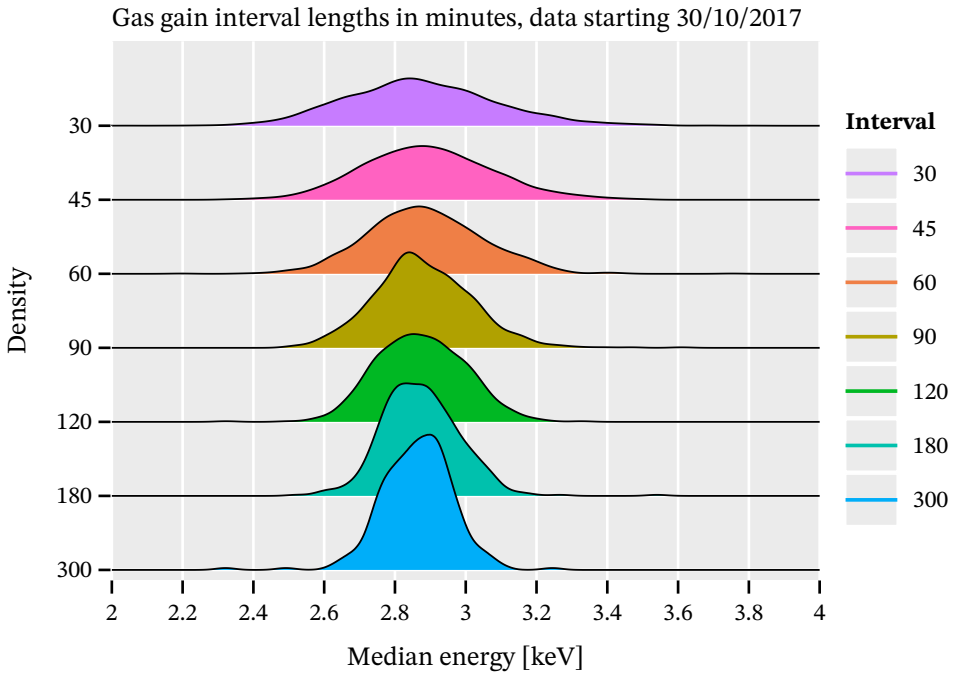
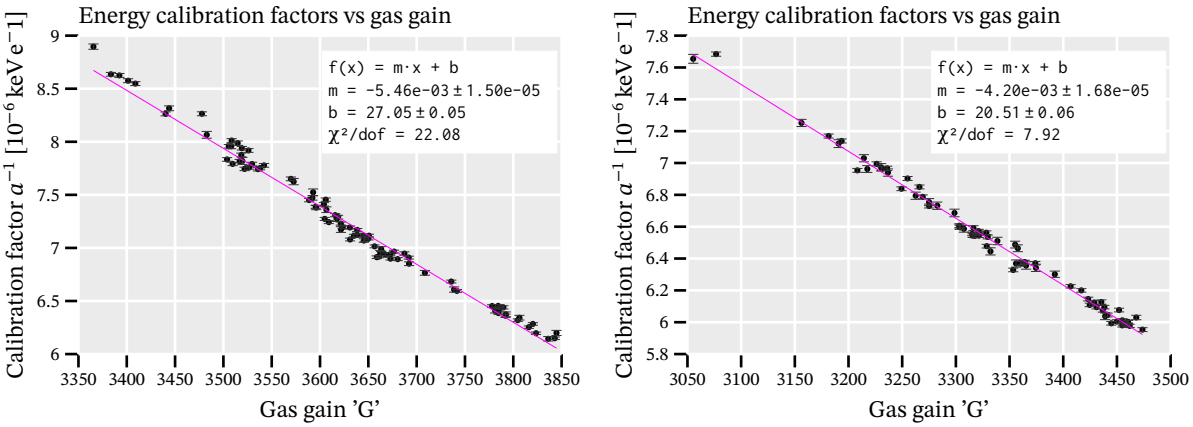


Figure 10.5.: Ridgeline plot of a kernel density estimation (bandwidth based on Silverman's rule of thumb) of the median cluster energies split by the used time intervals. The underlying data is the background data from Oct 2017 to Dec 2017. The overlap of the individual ridges is for easier visual comparison and a KDE was selected over a histogram due to strong binning dependence of the resulting histograms. For the dataset and binning the 90 min interval (olive) strikes an acceptable balance between unimodality and variance.

As the different ways to look at the data are not entirely conclusive, the choice was made to choose an interval length that is not too long, while still providing enough statistics for the ^{55}Fe spectra. As such 90 min was selected as the final interval time. Of course no data taking run is a perfect multiple of 90 min. The last slice smaller than the time interval is either added to the second to last slice (making it longer than 90 min) if it is smaller than some fixed interval or will be kept as a single shorter interval. This is controlled by an additional parameter that is set to 25 min by default. [4]

10.3. Energy calibration dependence on the gas gain

With the final choice of time interval for the gas gain binning in place, the actual calibration used for further analysis can be presented. Fig. 10.6 shows the fits according to the linear relation as explained in sec. 10.1, eq. 10.2, for the two data taking campaigns, Run-2 in 10.6(a) and Run-3 in 10.6(b). Each point represents one 90 min slice of calibration data for which a ^{55}Fe spectrum was fitted and then the linear energy calibration performed. The resulting energy calibration factor is then plotted against the gas gain computed for this time slice. The uncertainty of each point is the uncertainty extracted from the fit parameter of the calibration factor after error propagation. Calibrations need to be performed separately for each data taking campaign as the detector behavior changes due to different detector calibrations. These have an impact on the ToT calibration as well as the activation threshold. Note that the reduced χ^2 values shown in the figure, implies that this calibration does not perfectly calibrate out the systematic effects of the variability.



(a) Run-2

(b) Run-3

Figure 10.6.: Fit of the gas gain values vs the calculated energy calibration factors for all calibration runs in Run-2 (a) and Run-3 (b). Each run was further sliced into 90 min parts for the gas gain determination and ^{55}Fe fits.

[4] Both the gas gain time slicing and the minimum length for the last slice in a run to be kept as a shorter slice can be configured from the TPA configuration file, via the `gasGainInterval` and `minimumGasGainInterval` fields, respectively.

To compare the energy calibration using single gas gain values for each full run against the method of time slicing them to 90 min chunks, we will look at the median cluster energy in each time slice for background and calibration data. This is the same idea as behind fig. 10.2 previously, just for the energy instead of charge. This yields fig. 10.7. The points represent background and crosses calibration data. Green is the unbinned (full run) approach and purple the binned approach using 90 min slices. The effect is a slight, but visible reduction in variance. It represents an important aspect of increasing data reliability and lowering associated systematic uncertainties. Note that the variability looks much smaller than in fig. 10.2 due to not being normalized. However, here we wish to emphasize that the absolute energy calibration yields a flat result and matches our expectation.

As such the final energy calibration works by first deducing the gas gain at the time of an event, computing the calibration factor required for this gas gain and finally using that factor to convert the charge into energy.

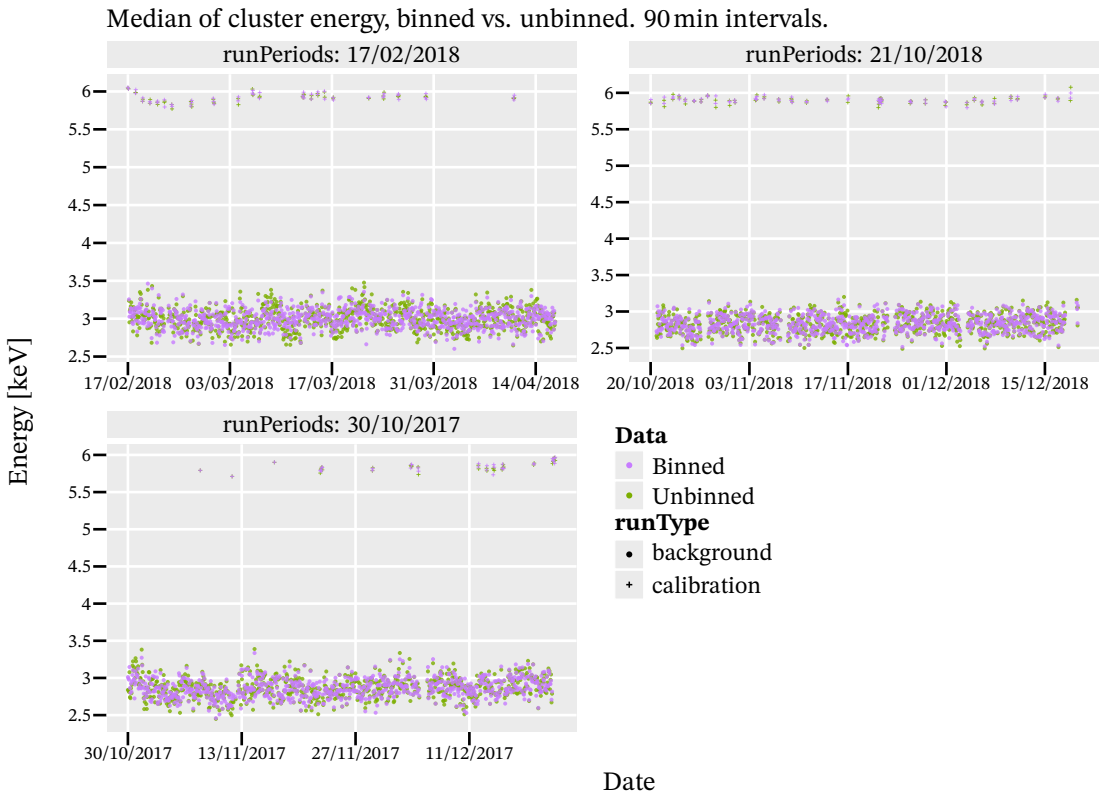


Figure 10.7.: Median of the cluster energy after calibration using two different approaches. Green corresponds to calculating the energy based on a single gas gain for each run and purple implies calculation based on 90 min time intervals for the gas gain. Both cases use the same 90 min intervals to compute a local, temporal median of all clusters. Each subplot corresponds to a data taking period with significant times between for clarity. The energies from unbinned gas gains has a wider distribution than the binned data. The latter approaches a flat distribution of the background energies (points) better than the former. The impact for the calibration data (crosses) is much smaller, as they are not much longer than the 90 min binning anyway.

10.4. FADC

As touched on multiple times previously, in particular in sec. 9.5, the FADC suffered from noise. This meant multiple changes to the amplifier settings to mitigate this.

We will now go over what the FADC noise looks like and explain our noise filter used to determine if an FADC event is noisy (to ignore it), in sec. 10.4.1. Then we check the impact of the different amplification settings on the FADC data and discuss the impact of the FADC data quality on the rest of the detector data in sec. 10.4.2.

10.4.1. FADC noise example and detection

An example of the most common type of noise events seen in the FADC data is shown in fig. 10.8. As the FADC registers effectively correspond to 1 ns time resolution, the periodicity of these noise events is about 150 ns, corresponding to roughly 6.6 MHz frequency. Other types of less common noise events are events with a noise frequency of about 1.5 MHz. [5] A final type of noise events are events in which the FADC input is fully at a low input voltage (in the tens of mV range), but contains no real 'activity'. The values though are lower than the threshold in these cases triggering the FADC.

For data analysis purposes, in particular when the FADC data is used in conjunction with the GridPix data, it is important to not accidentally use an FADC event, which contains noise. While generally noise events are unlikely to be part of physical ionization events on the center GridPix it is better to be on the conservative side. The noise analysis is kept very simple [6]. The FADC spectrum, consisting of 2560 registers, is being sliced into 150 register wide intervals. In each interval we check for the minimum of the signal, m_s. The slice is adjusted around the found minimum to check if the minimum is contained fully in the slice (if not it is part of the next slice). If that minimum is below $m_s < B - \sigma$, where B is the signal baseline and σ the standard deviation of the full FADC signal (including the peaks!), it will be counted as a peak. The noise filter detection is then defined by signals with at least 4 peaks within slices of 150 ns.

10.4.2. Amplifier settings impact and activation threshold

Now let's look at the impact of the different amplifier settings on the FADC data properties. This includes differences in the rise time and fall time, but because changing the integration and differentiation times on the amplifier has a direct impact on the absolute amplification on the signal, we also need to consider the change in activation threshold of the signals.

[5] The frequencies are on the low end in terms of common radio communication frequencies. The leading assumption has always been that the source likely points to noise produced by e.g. the motors moving the CAST magnet and similar.

[6] I started with a simple implementation, intending to replace it later. But it worked well enough that I simply kept it so far.

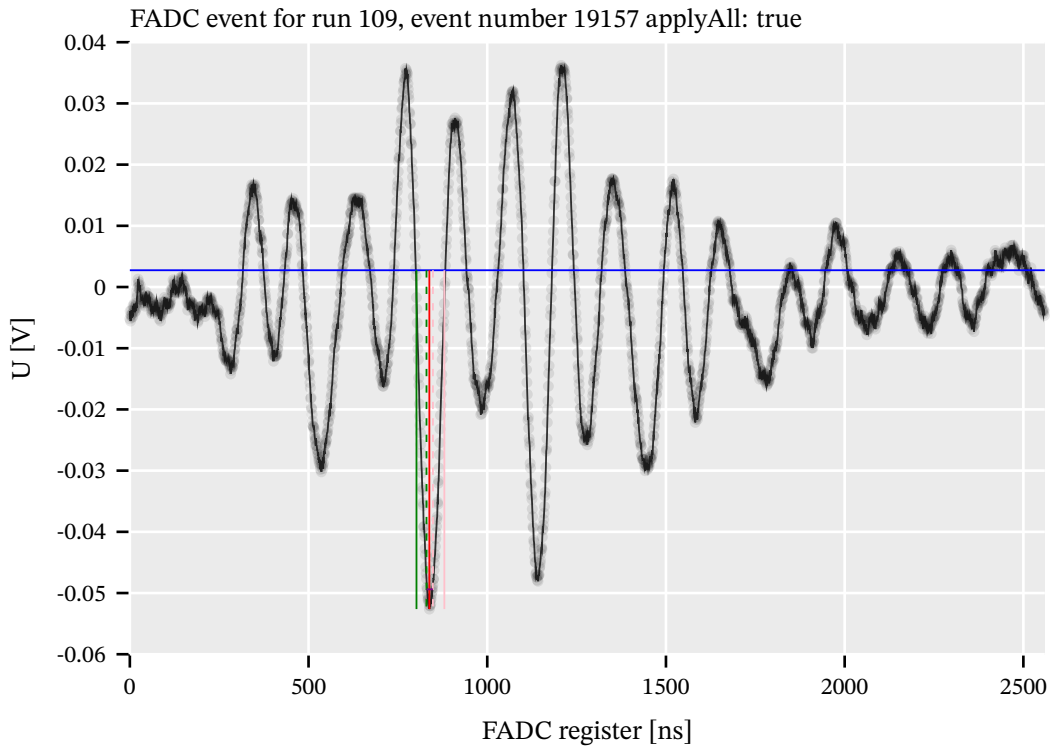


Figure 10.8.: Example of the most common type of noise example. Noise has a periodicity of about $\sim$ 150 ns, or about 6.6 MHz.

As a short reminder, the FADC settings were changed twice during the Run-2 period in 2017. Starting from 2018, the settings were left unchanged from the end of 2017. An overview of the setting changes is shown in tab. 10.1. Note that the Ortec amplifier has a coarse and a fine gain. Only the coarse gain was changed. [7] The gain changes were performed to counteract the resulting amplification changes due to the integration and differentiation setting changes (this is documented as a side effect in the FADC manual [55]).

Runs	Integration [ns]	Differentiation [ns]	Coarse gain
76 - 100	50	50	6x
101 - 120	50	20	10x
121 - 306	100	20	10x

Table 10.1.: Overview of the different FADC amplifier settings and the associated run numbers.

The ^{55}Fe calibration spectra come in handy for the FADC data, as they also give a known baseline to compare against for this type of data. To get an idea of the rise and fall times of the FADC for different settings, we can compute a truncated mean of all rise and fall times in each calibration run.

[7] At least to my memory and notes, which should otherwise contain that.

[8] This is done in fig. 10.9, which shows the mean rise time for each run in 2017 with the fall time color coded in each point. The shaded regions indicate the FADC amplifier settings. Changing the differentiation time from 50 ns down to 20 ns decreased the rise time by about 10 ns. The change in the fall time is much more pronounced. The change of the integration time from 50 ns to 100 ns then brings the rise time back up by about 5 ns with now a drastic extension in the fall time from the mid 200 ns to over 400 ns. Clearly the fall time is much more determined by the amplifier settings.

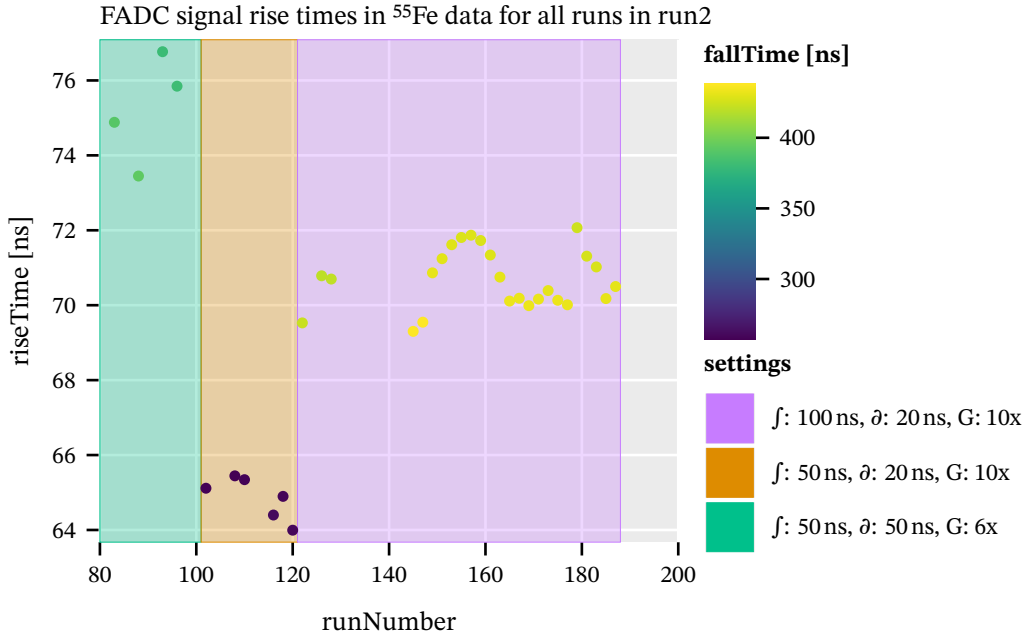


Figure 10.9.: The mean rise time of the FADC signals recorded during the ^{55}Fe data during Run-2 of the FADC. Again, the FADC amplifier settings are visible as expected. '$\int$': integration time, '∂': differentiation time, 'G': coarse gain.

A direct scatter plot of the rise times against the fall times is shown in fig. 10.10, where the drastic changes to the fall time are even more pronounced. Each point once again represents one ^{55}Fe calibration run. The different settings manifest as separate 'islands' in this space.

The FADC pulses contain a measure of the total charge that was induced on the grid and therefore an indirect measure of the charge seen on the center GridPix. The ^{55}Fe calibration runs could be used to fully calibrate the FADC signals in charge if desired. Ideally one would fully (numerically) integrate the FADC signal for each event to compute an effective charge. As we only use the FADC signals in the context of this thesis for their sensitivity to longitudinal shape information, this is not implemented. For ^{55}Fe calibration data the amplitude of the FADC pulse is a direct proxy for the charge anyway, because the signal shape is (generally) the same for X-rays. [9] For the determination

[8] We use a truncated mean of all data within the 5$^{\text{th}}$ and 95$^{\text{th}}$ percentile if the rise and fall time values. This is just to make the numbers less susceptible to extreme outliers. Alternatively, we could of course also look at the median for example.

[9] As with everything, this is only an approximation and completely neglects possible nonlinearities in amplitude vs. inte-

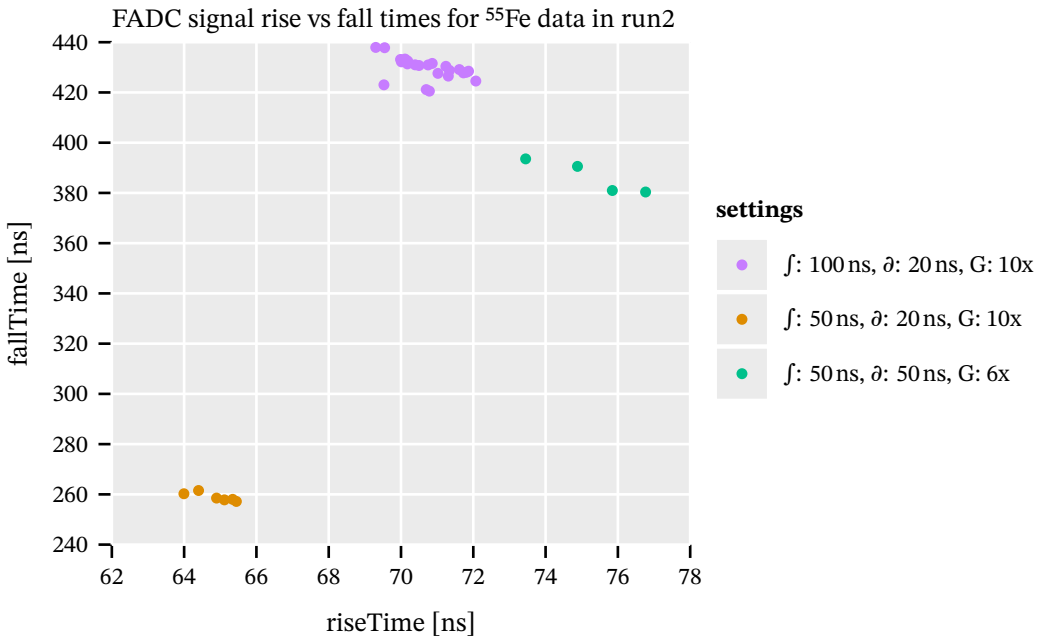


Figure 10.10.: The mean rise time of the FADC signals recorded during the ^{55}Fe data against the fall time during Run-2 at CAST. One point for each calibration run. The different settings create three distinct blobs. '∫': integration time, '∂': differentiation time, 'G': coarse gain.

of whether the gas gain variations discussed in sec. 10.2 have a physical origin due to changing gas gain or are caused by electronic effects, we already included the FADC data in fig. 10.3 of sec. 10.2.1. Computing the histogram of all amplitudes of the FADC signals in a ^{55}Fe calibration run, yields a ^{55}Fe spectrum like for the center GridPix. The fitted position of the photopeak in these spectra is then a direct counterpart to those computed for the GridPix. Due to its independence and only being sensitive to induced charge, it acts as a good validator.

In the context of the FADC amplifier settings it is interesting to see how the photopeak position changes between runs when computed like that. This is shown in fig. 10.11. We can see that the initial change in differentiation time resulted in a larger gain decrease than the attempt at compensation from `6x` to `10x` on the coarse gain. The final increase in the integration time then caused another drop in signal amplitudes, implying an even lower absolute gain. In addition though the gas gain variation within a single setting is very visible (as discussed in sec. 10.2.1).

Finally, we can look at the activation threshold of the FADC. The easiest way to do this is the following: we read the energies of all events on the center GridPix, then map them to their FADC events. Although not common in calibration data, some events may not trigger the FADC. By then computing – for example – the first percentile of the energy data (the absolute lowest value may be some outlier) we automatically get the lowest equivalent GridPix energy that triggers the FADC.

gral and so on. But it works well for its purpose here.

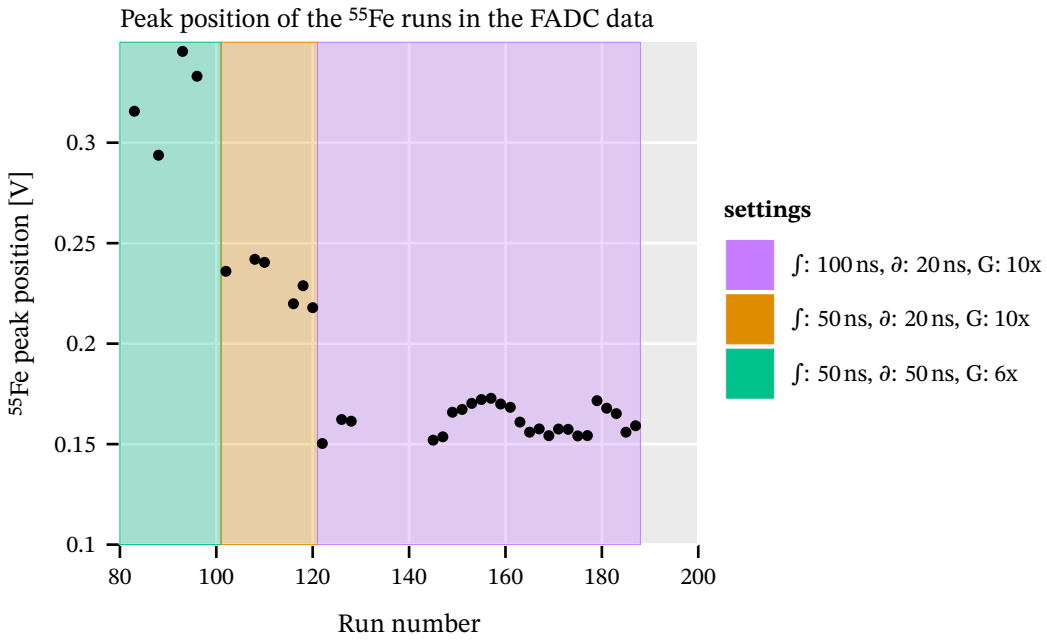


Figure 10.11.: The peak position in V of the photopeak in the ^{55}Fe calibration runs during Run-2 as seen on the FADC. The different FADC amplifier settings are clearly visible. '$\int$': integration time, '∂': differentiation time, 'G': coarse gain.

Doing this leads to yet another similar plot to the previous, fig. 10.12. With the first FADC settings the activation threshold was at a low $\sim$ 1.1 keV. Unfortunately, both amplifier settings moved the threshold further up to about 2.2 keV with the final settings. In hindsight it likely would have been a better idea to try to run with a lower activation threshold so that the FADC trigger is available for more events at low energies. However, at the time of the data taking campaign not all information was available for an educated assessment, nor was there enough time to test and implement other ideas. Especially because there is a high likelihood that other settings might have run back into noise problems.

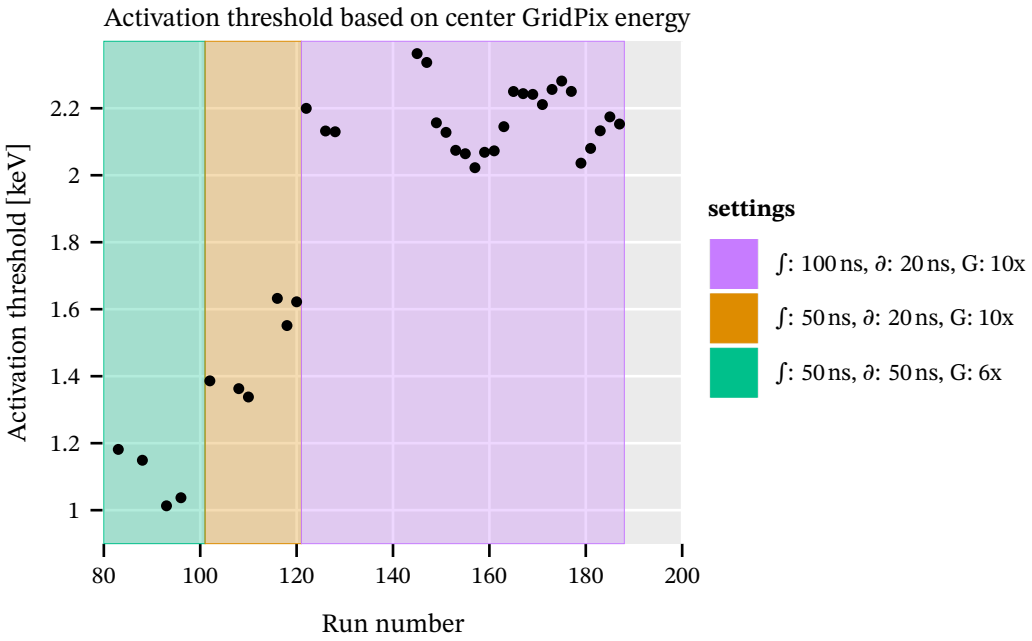


Figure 10.12.: The activation threshold of the FADC for each calibration run in 2017. Computed by the first percentile of the corresponding energies recorded by the GridPix. '∫': integration time, '∂': differentiation time, 'G': coarse gain.

11. Finding signal and defining background | Background rate computation Analysis

Contents

With the CAST data fully reconstructed and energy calibrated, it is time to define the methods used to extract axion candidates and derive a background rate from the data. We first introduce the likelihood cut method in sec. 11.1 to motivate the need for reference X-ray data from an X-ray tube. Such data, taken at the CAST Detector Lab (CDL) will be discussed in depth in sec. 11.2. We then see how the reference data is used in the likelihood method to act as an event classifier in sec. 11.3. As an alternative to the likelihood cut, we will introduce another classifier in the form of a simple artificial neural network in sec. 11.4. This is another in depth discussion, as the selection of the training data and verification is non-trivial. With both classifiers discussed, it is time to include all other Septemboard detector features as additional vetoes in sec. 11.5. At the very end we will look at background rates for different cases, sec. 11.6, motivating the approach of our limit calculation in the next chapter.

> Note: The methods discussed in this chapter are generally classifiers that predict how 'signal-like' a cluster is. Based on this prediction we will usually define a cut value as to keep a cluster as a potential signal. This means that if we apply the method to background data (that is, CAST data taken outside of solar trackings) we recover the 'background rate'; the irreducible amount of background left (at a certain efficiency), which is signal-like. If instead we apply the same methods to CAST solar tracking data we get instead a set of 'axion induced X-ray candidates'. [1] In the context of the chapter we commonly talk about "background rates", but the equivalent meaning in terms of tracking data and candidates should be kept in mind.

11.1. Likelihood method

The detection principle of the GridPix detector implies physically different kinds of events will have different geometrical shapes. An example can be seen in fig. 11.1(a), comparing the cluster eccentricity of ^{55}Fe calibration events with background data. This motivates usage of a geometry based approach to determine likely signal-like or background-like clusters. The method to distinguish the two types of events is a likelihood cut, based on the one in [134]. It effectively assigns a single value to each cluster for the likelihood that it is a signal-like event.

[1] Of course the set of candidates contains background itself. The terminology 'candidate' intends to communicate that each candidate may be a background event or potentially a signal due to axions. But that is part of chapter 12.

Specifically, this likelihood method is based on three different geometric properties (also see sec. 8.4.2):

1. the eccentricity ε of the cluster, determined by computing the long and short axis of the two dimensional cluster and then computing the ratio of the RMS of the projected positions of all active pixels within the cluster along each axis.

2. the fraction of all pixels within a circle of the radius of one transverse RMS from the cluster center, f.

3. the length of the cluster (full extension along the long axis) divided by the transverse RMS, l.

These variables are obviously highly correlated, but still provide a very good separation between the typical shapes of X-rays and background events. They mainly characterize the ”spherical-ness” as well as the density near the center of the cluster, which is precisely the intuitive sense in which these type of events differ. For each of these properties we define a probability density function $\mathcal{P}_i$, which can then be used to define the likelihood of a cluster with properties (ε, f, l) to be signal-like:

$$\mathcal{L}(\varepsilon, f, l) = \mathcal{P}_\varepsilon(\varepsilon) \cdot \mathcal{P}_f(f) \cdot \mathcal{P}_l(l) \tag{11.1}$$

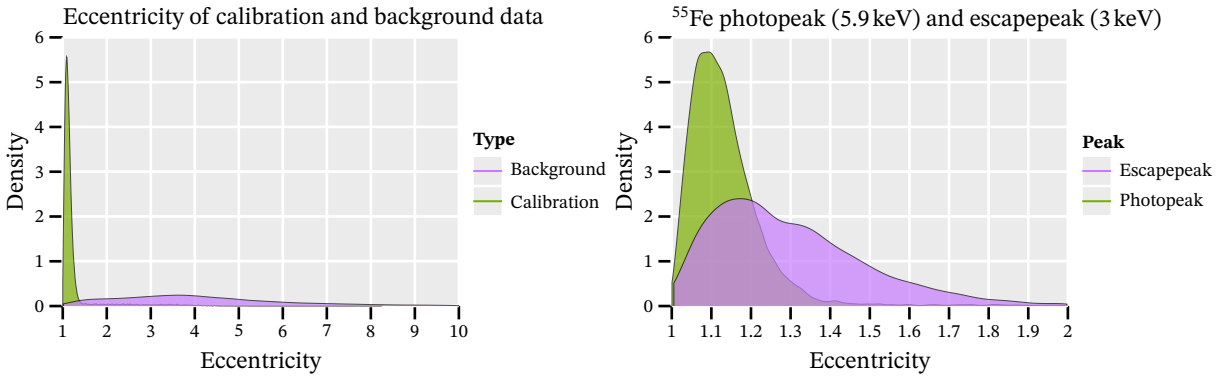
where the subscript is denoting the individual probability density and the argument corresponds to the individual value of each property.

This raises the important question of what defines each individual probability density $\mathcal{P}_i$. In principle it can be defined by computing a normalized density distribution of a known dataset, which contains representative signal-like data. The ^{55}Fe calibration data from CAST contains such representative data, if not for one problem: the properties used in the likelihood method are energy dependent, as seen in fig. 11.1(b), a comparison of the eccentricity of X-rays from the photopeak of the ^{55}Fe calibration source compared to those from the escape peak. The CAST calibration data can only characterize two different energies, but the expected axion signal is a (model dependent) continuous spectrum.

For this reason data was taken using an X-ray tube with 8 different target / filter combinations to provide the needed data to compute likelihood distributions for X-rays of a range of different energies. The details will be discussed in the next section, 11.2.

11.2. CAST Detector Lab

In this section we will cover the X-ray tube data taken at the CAST Detector Lab (CDL) at CERN. First, we will show the setup in sec. 11.2.1. Next the different target / filter combinations that were used will be discussed and the measurements presented in sec. 11.2.2. These measurements then



(a) Calibration & background

(b) Photopeak & escape peak

Figure 11.1.: (a)Comparison of the eccentricity of clusters from a calibration and a background run. Background events are clearly much more eccentric on average. (b)The X-ray properties are energy dependent as can be seen in the comparison between the eccentricity of X-rays from the photopeak compared to those from the escape peak. A kernel density estimation is used in both figures.

are used to define the probability densities for the likelihood method, see sec. 11.2.5. Further, in sec. 11.2.6, we cover a few more details on a linear interpolation we perform to compute a likelihood distribution at an arbitrary energy. And finally, they can be used to measure the energy resolution of the detector at different energies, sec. 11.2.8.

The data presented here was also part of the master thesis of Hendrik Schmick [189]. Further, the selection of lines and approach follows the ideas used for the single GridPix detector in [134] with some notable differences. For a reasoning for one particular difference in terms of data treatment, see appendix H.

> Note: in the course of the CDL related sections the term target/filter combination (implicitly including the used high voltage setting) is used interchangeably with the main fluorescence line (or just 'the fluorescence line') targeted in a specific measurement. Be careful while reading about applied high voltages in kV and energies in keV. The produced fluorescence lines typically have about 'half' the energy in keV as the applied voltage in kV. See tab. 11.3 in sec. 11.2.2 for the precise relation.

11.2.1. CDL setup

The CAST detector lab provides a vacuum test stand, which contains an X-ray tube. A Micromegas-like detector can easily be mounted to the rear end of the test stand. An X-ray tube uses a filament to produce free electrons, which are then accelerated with a high voltage of the order of multiple kV. The main part of the setup is a rotateable wheel inside the vacuum chamber, which contains a set of 18 different positions with 8 different target materials, as seen in tab. 11.1. The highly energetic

electrons interacting with the target material generate a continuous Bremsstrahlung spectrum with characteristic lines depending on the target. A second rotateable wheel contains a set of 11 different filters, see tab. 11.2. These can be used to filter out undesired parts of the generated spectrum by choosing a filter that is opaque in the those energy ranges.

As mentioned previously in sec. 9.1, the detector was dismounted from CAST in Feb. 2019 and installed in the CAST detector lab on <*2019-02-14 Thu*>. The week from <*2019-02-15 Fri*> to <*2019-02-21 Thu*> X-ray tube data was taken in the CDL.

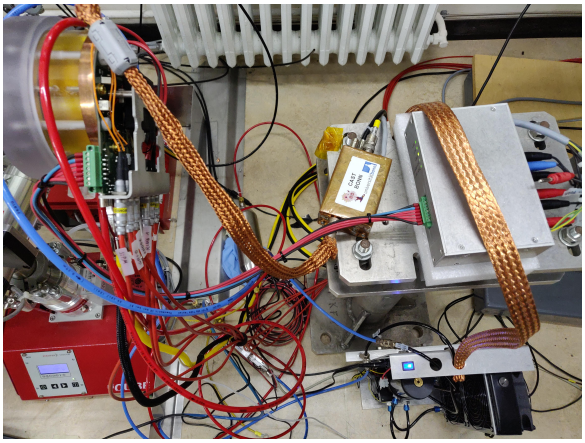
Fig. 11.2(a) shows the whole vacuum test stand, which contains the X-ray tube on the front left end and the Septemboard detector installed on the rear right, visible by the red HV cables and yellow HDMI cable. In fig. 11.2(b) we see the whole detector part from above, with the Septemboard detector installed to the vacuum test stand on the left side. The water cooling system is seen in the bottom right, with the power supply above that. The copper shielded box slightly right of the center is the Ortec pre-amplifier of the FADC, which is connected via a heavily shielded LEMO cable to the Septemboard detector. This cable was available in the CAST detector lab and proved invaluable for the measurements as it essentially removed any noise visible in the FADC signals (compared to the significant noise issues encountered at CAST, sec. 9.5.1). Given the activation threshold of the FADC with CAST amplifier settings (see sec. 10.4.2) at around 2 keV, the amplification needed to be adjusted in the CDL on a per-target basis. The shielded cable allowed the FADC to even act as a trigger for 277 eV $C_{K\alpha}$ X-rays without any noise problems. [2]

Target Material	Position
Ti	1
Ag	2
Mn	3
C	4
BN	5
Au	6
Al	7 - 11
Cu	12 - 18

Table 11.1.: Table of all available target materials in the CAST detector lab and their respective position on a rotateable wheel.

[2] In hindsight it seems obvious that a similarly shielded LEMO cable should have been used for the measurements at CAST. Unfortunately, due to lack of experience with electromagnetic interference this was not considered before.

(a) Full setup



(b) Detector setup

Figure 11.2.: (a) shows the full vacuum test stand containing the X-ray tube with the Septemboard detector installed at the rear, visible by the red HV and yellow HDMI cables. (b) is a view of the detector setup from above. On the left hand side is the detector mounted to the vacuum setup. The water cooling is seen in the bottom right, connected via the blue tubes. The gas supply is in red tubing and the power supply is visible on the right above the water cooling (with a green Phoenix connector). The copper shielded cable is a LEMO cable for the FADC signal going to the pre-amplifier (with the University of Bonn sticker).

Filter material	Position
Ni 0.1 mm	1
Al 5 µm	2
PP G12 (EPIC)	3
PP G12 (EPIC)	4
Cu 10 µm	5
Ag 5 µm (99.97%) AG000130/24	6
Fe 50 µm	7
Mn foil 50 µm (98.7% + permanent polyester)	8
Co 50 µm (99.99%) CO0000200	9
Cr 40 µm (99.99% + permanent polyester)	10
Ti 50 µm (99.97%) TI000315	11

Table 11.2.: Table of all available filters in the CAST detector lab and their respective position on a rotateable wheel.

11.2.2. CDL measurements

The measurements were performed with 8 different target and filter combinations, with a higher density towards lower energies due to the nature of the expected solar axion flux. For each target and filter at least two runs were measured, one *with* and one *without* using the FADC as a trigger. The latter was taken to collect more statistics in a shorter amount of time, as the FADC readout slows down the data taking speed due to increased dead time. Table 11.3 provides an overview of all data taking runs, the target, filter and HV setting, the main X-ray fluorescence line targeted and its energy. Finally, the mean position of the main fluorescence line in the charge spectrum and its width as determined by a fit is shown. As can be seen the position moves in some cases significantly between different runs, for example in case of the Cu − Ni measurements. Also within a single run some stronger variation can be seen, evident by the much larger line width for example in run 320 compared to 319. See appendix sec. G.2 for all measured spectra (pixel and charge spectra), where each plot contains all runs for a target / filter combination. For example fig. G.8(b) shows runs 319, 320 and 345 of the Cu − Ni measurements at 15 kV with the strong variation in charge position of the peak visible between the runs.

The variability both between runs for the same target and filter as well as within a run shows the detector was undergoing gas gain changes similar to the variations at CAST (sec. 10.2). With an understanding that the variation is correlated to the temperature this can be explained. The small laboratory underwent significant temperature changes due to the presence of 3 people, all running machines and freely opening and closing windows, in particular due to very warm temperatures relative to a typical February in Geneva. [3] Fig. 11.3 shows the calculated gas gains (based on 90 min intervals) for each run colored by the target/filter combination. It undergoes a strong, almost exponential change over the course of the data taking campaign. Because of this variation between runs, each run is treated fully separately in contrast to [134], where all runs for one target / filter combination were combined.

Fortunately, this significant change of gas gain does not have an impact on the distributions of the cluster properties. See appendix H.1 for comparisons of the cluster properties of the different runs (and thus different gas gains) for each target and filter combination.

As the main purpose is to use the CDL data to generate reference distributions for certain cluster properties, the relevant clusters that correspond to known X-ray energies must be extracted from the data. This is done in two different ways:

[3] The measurement campaign at the CDL was done in February. However, the weather was nice and sunny most of the week, if I remember correctly. The laboratory has windows towards the south-east. With reasonably cold outside temperatures and sunshine warming the lab in addition to machines and people, opening and closing windows changed the temperature on short time scales, likely significantly contributing to the detector instability. In the extended version of this thesis you'll find a section below this one containing the weather data at the time of the CDL data taking.

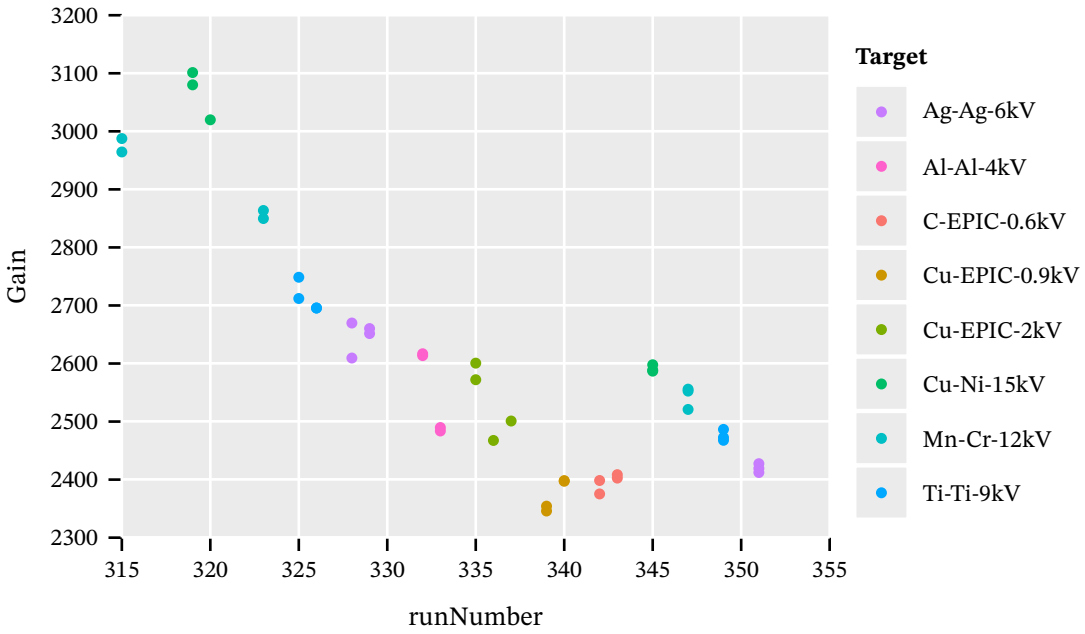


Figure 11.3.: Gas gain of each gas gain slice of 90 min in all CDL runs. A significant decrease over the runs (equivalent to the week of data taking) is visible. Multiple points for the same run correspond to multiple gas gain time slices as explained in sec. 10.2.2.

1. A set of fixed cuts (one set for each target/filter combination) is applied to each run, as presented in tab. 11.4. This is the same set as used in [134]. Its main purpose is to remove events with multiple clusters and potential background contributions.

2. By cutting around the main fluorescence line in the charge spectrum in a 3σ region, for which the spectrum needs to be fitted with the expected lines, see sec. 11.2.3. This is done on a run-by-run basis.

The remaining data after both sets of cuts can then be combined for each target/filter combination to make up the distributions for the cluster properties as needed for the likelihood method.

For a reference of the X-ray fluorescence lines (for more exact values and α_1, α_2 values etc.) see tab. 5.1. The used EPIC filter refers to a filter developed for the EPIC camera of the XMM-Newton telescope. It is a bilayer of 1600 Å polyimide and 800 Å aluminium. For more information about the EPIC filters see references [212, 216, 34, 35], in particular [35] for an overview of the materials and production.

Run	FADC?	Target	Filter	HV [kV]	Line	Energy [keV]	μ [e$^-$]	σ [e$^-$]	σ/μ
319	y	Cu	Ni	15	Cu K_α	8.04	$9.509(21) \times 10^5$	$7.82(18) \times 10^4$	$8.22(19) \times 10^{-2}$
320	n	Cu	Ni	15			$9.102(22) \times 10^5$	$1.010(19) \times 10^5$	$1.110(21) \times 10^{-1}$
345	y	Cu	Ni	15			$6.680(12) \times 10^5$	$7.15(11) \times 10^4$	$1.070(16) \times 10^{-1}$
315	y	Mn	Cr	12	Mn K_α	5.89	$6.321(29) \times 10^5$	$9.44(26) \times 10^4$	$1.494(41) \times 10^{-1}$
323	n	Mn	Cr	12			$6.328(11) \times 10^5$	$7.225(89) \times 10^4$	$1.142(14) \times 10^{-1}$
347	y	Mn	Cr	12			$4.956(10) \times 10^5$	$6.211(82) \times 10^4$	$1.253(17) \times 10^{-1}$
325	y	Ti	Ti	9	Ti K_α	4.51	$4.83(31) \times 10^5$	$4.87(83) \times 10^4$	$1.01(18) \times 10^{-1}$
326	n	Ti	Ti	9			$4.615(87) \times 10^5$	$4.93(25) \times 10^4$	$1.068(57) \times 10^{-1}$
349	y	Ti	Ti	9			$3.90(23) \times 10^5$	$4.57(57) \times 10^4$	$1.17(16) \times 10^{-1}$
328	y	Ag	Ag	6	Ag L_α	2.98	$3.0682(97) \times 10^5$	$3.935(79) \times 10^4$	$1.283(26) \times 10^{-1}$
329	n	Ag	Ag	6			$3.0349(51) \times 10^5$	$4.004(40) \times 10^4$	$1.319(13) \times 10^{-1}$
351	y	Ag	Ag	6			$2.5432(63) \times 10^5$	$3.545(49) \times 10^4$	$1.394(20) \times 10^{-1}$
332	y	Al	Al	4	Al K_α	1.49	$1.4868(50) \times 10^5$	$2.027(38) \times 10^4$	$1.364(26) \times 10^{-1}$
333	n	Al	Al	4			$1.3544(30) \times 10^5$	$2.539(24) \times 10^4$	$1.875(18) \times 10^{-1}$
335	y	Cu	EPIC	2	Cu L_α	0.930	$8.885(99) \times 10^4$	$1.71(11) \times 10^4$	$1.93(13) \times 10^{-1}$
336	n	Cu	EPIC	2			$7.777(94) \times 10^4$	$2.39(14) \times 10^4$	$3.08(19) \times 10^{-1}$
337	n	Cu	EPIC	2			$7.86(15) \times 10^4$	$2.47(11) \times 10^4$	$3.14(15) \times 10^{-1}$
339	y	Cu	EPIC	0.9	O K_α	0.525	$5.77(11) \times 10^4$	$1.38(22) \times 10^4$	$2.39(39) \times 10^{-1}$
340	n	Cu	EPIC	0.9			$4.778(31) \times 10^4$	$1.230(50) \times 10^4$	$2.58(11) \times 10^{-1}$
342	y	C	EPIC	0.6	C K_α	0.277	$4.346(36) \times 10^4$	$1.223(29) \times 10^4$	$2.814(70) \times 10^{-1}$
343	n	C	EPIC	0.6			$3.952(20) \times 10^4$	$1.335(14) \times 10^4$	$3.379(40) \times 10^{-1}$

Table 11.3.: Overview of all runs taken behind the X-ray tube, whether they ran with or without FADC, their targets, filters and high voltage setting and information about the major fluorescence line and its energy. Finally, the mean μ and width σ of the main fluorescence line as determined by the fit plus the resulting energy resolution μ/σ is shown.

Target	Filter	line	HV	length	RMS_{Tmin}	RMS_{Tmax}	eccentricity
Cu	Ni	Cu K_α	15		0.1	1.0	1.3
Mn	Cr	Mn K_α	12		0.1	1.0	1.3
Ti	Ti	Ti K_α	9		0.1	1.0	1.3
Ag	Ag	Ag L_α	6	6.0	0.1	1.0	1.4
Al	Al	Al K_α	4		0.1	1.1	2.0
Cu	EPIC	Cu L_α	2		0.1	1.1	2.0
Cu	EPIC	O K_α	0.9		0.1	1.1	2.0
C	EPIC	C K_α	0.6	6.0	0.1	1.1	

Table 11.4.: Cuts applied to the CDL datasets in order to roughly clean of potential background events and double hits (recognized as single cluster). RMS refers to the transverse RMS of the clusters.

11.2.3. Charge spectra of the CDL data

To the spectra of each run of the charge data a mixture of Gaussian functions is fitted. [4]

Specifically the Gaussian expressed as

$$G(E; \mu, \sigma, N) = \frac{N}{\sqrt{2\pi}} \exp\left(-\frac{(E-\mu)^2}{2\sigma^2}\right), \tag{11.2}$$

is used and will be referenced as G with possible arguments from here on. Note that while the physical X-ray transition lines are Lorentzian shaped [224], the lines as detected by a gaseous detector are entirely dominated by detector resolution, resulting in Gaussian lines. For other types of detectors used in X-ray fluorescence (XRF) analysis, convolutions of Lorentzian and Gaussian distributions are used [111, 104], called pseudo Voigt functions [181, 98].

The functions fitted to the different spectra then depend on which fluorescence lines are visible. The full list of all combinations is shown in tab. 11.5. Typically each line that is expected from the choice of target, filter and chosen voltage is fitted, if it can be visually identified in the data. [5] If no 'argument' is shown in the table to G it means each parameter (N, μ, σ) is fitted. Any specific argument given implies that parameter is fixed relative to another parameter. For example $\mu_{L\alpha}^{\text{Ag}} \cdot \left(\frac{3.151}{2.984}\right)$ fixes the $L\beta$ line of silver to the fit parameter $\mu_{L\alpha}^{\text{Ag}}$ of the $L\alpha$ line with a multiplicative shift based on the relative eneries of $L\alpha$ to $L\beta$. In some cases the amplitude is fixed between different lines where relative amplitudes cannot be easily predicted or determined, e.g. in one of the Cu − EPIC runs, the $C_{K\alpha}$ line is fixed to a tenth of the $O_{K\alpha}$ line. This is done to get a good fit based on trial and error.

[4] Note that similar fits can be performed for the pixel spectra as well. However, as these are not needed for anything further they are only presented in the extended version of the thesis.

[5] The restriction to what is 'visually identifiable' is to avoid the need to fit lines lost in detector resolution and signal to noise ratio. In addition, it avoids overfitting to large numbers of parameters (which brings its own challenges and problems).

Finally, in both Cu − EPIC lines two 'unknown' Gaussians are added to cover the behavior of the data at higher charges. The physical origin of this additional contribution is not entirely clear. The used EPIC filter contains an aluminum coating [35]. As such it has the aluminum absorption edge at about 1.5 keV, possibly matching the additional contribution for the 2 kV dataset. Whether it is from a continuous part of the spectrum or a form of aluminum fluorescence is not clear however. This explanation does not work in the 0.9 kV case, which is why the line is deemed 'unknown'. It may also be a contribution of the specific polyimide used in the EPIC filter [35]. Another possibility is it is a case of multi-cluster events, which are too close to be split, but with properties close enough to a single X-ray as to not be removed by the cleaning cuts (which gets more likely the lower the energy is).

The full set of all fits (including the pixel spectra) is shown in appendix G.2. Fig. 11.4 shows the charge spectrum of the Ti target and Ti filter at 9 kV for one of the runs. These plots show the raw data in the green histogram and the data left after application of the cleaning cuts (tab. 11.4) in the purple histogram. The black line is the result of the fit as described in tab. 11.5 with the resulting parameters shown in the box (parameters that were fixed are not shown). The black straight lines with grey error bands represent the 3σ region around the main fluorescence line, which is used to extract those clusters likely from the fluorescence line and therefore known energy.

Target	Filter	HV [kV]	Fit functions
Cu	Ni	15	$G^{\text{Cu}}_{K\alpha} + G^{\text{Cu,esc}}_{K\alpha}$
Mn	Cr	12	$G^{\text{Mn}}_{K\alpha} + G^{\text{Mn,esc}}_{K\alpha}$
Ti	Ti	9	$G^{\text{Ti}}_{K\alpha} + G^{\text{Ti,esc}}_{K\alpha} + G^{\text{Ti}}_{K\beta}\left(\mu^{\text{Ti}}_{K\alpha}\cdot\left(\frac{4.932}{4.511}\right), \sigma^{\text{Ti}}_{K\alpha}\right) + G^{\text{Ti,esc}}_{K\beta}\left(\mu^{\text{Ti}}_{K\alpha}\cdot\left(\frac{1.959}{4.511}\right), \sigma^{\text{Ti,esc}}_{K\alpha}\right)$
Ag	Ag	6	$G^{\text{Ag}}_{L\alpha} + G^{\text{Ag}}_{L\beta}\left(N^{\text{Ag}}_{L\alpha}\cdot 0.56, \mu^{\text{Ag}}_{L\alpha}\cdot\left(\frac{3.151}{2.984}\right), \sigma^{\text{Ag}}_{L\alpha}\right)$
Al	Al	4	$G^{\text{Al}}_{K\alpha}$
Cu	EPIC	2	$G^{\text{Cu}}_{L\alpha} + G^{\text{Cu}}_{L\beta}\left(N^{\text{Cu}}_{L\alpha}\cdot\left(\frac{0.65}{1.11}\right), \mu^{\text{Cu}}_{L\alpha}\cdot\left(\frac{0.9498}{0.9297}\right), \sigma^{\text{Cu}}_{L\alpha}\right) + G^{\text{O}}_{K\alpha}\left(\frac{N^{\text{Cu}}_{L\alpha}}{3.5}, \mu^{\text{Cu}}_{L\alpha}\cdot\left(\frac{0.5249}{0.9297}\right), \frac{\sigma^{\text{Cu}}_{L\alpha}}{2.0}\right) + G_{\text{unknown}}$
Cu	EPIC	0.9	$G^{\text{O}}_{K\alpha} + G^{\text{C}}_{K\alpha}\left(\frac{N^{\text{O}}_{K\alpha}}{10.0}, \mu^{\text{O}}_{K\alpha}\cdot\left(\frac{277.0}{524.9}\right), \sigma^{\text{O}}_{K\alpha}\right) + G_{\text{unknown}}$
C	EPIC	0.6	$G^{\text{C}}_{K\alpha} + G^{\text{O}}_{K\alpha}\left(\mu^{\text{C}}_{K\alpha}\cdot\left(\frac{0.525}{0.277}\right), \sigma^{\text{C}}_{K\alpha}\right)$

Table 11.5.: All fit functions for the charge spectra used for each target / filter combination. Typically each line that is expected and visible in the data is fit. G is a normal Gaussian. No 'argument' to G means each parameter (N, μ, σ) is fit. Specific arguments imply this parameter is fixed relative to another parameter, e.g. $\mu^{\text{Ag}}_{L\alpha}\cdot\left(\frac{3.151}{2.984}\right)$ fixes $L\beta$ line of silver to the fit parameter $\mu^{\text{Ag}}_{L\alpha}$ of $L\alpha$ with a multiplicative shift based on the relative eneries of $L\alpha$ to $L\beta$. In some cases the amplitude is fixed between different lines, e.g. in one of the Cu − EPIC runs the $\text{C}_{K\alpha}$ line is fixed to a tenth of the $\text{O}_{K\alpha}$ line. In both Cu − EPIC lines two 'unknown' Gaussians are added to cover the behavior of the data at higher charges. It is unclear what the real cause is, in particular in the lower energy case.

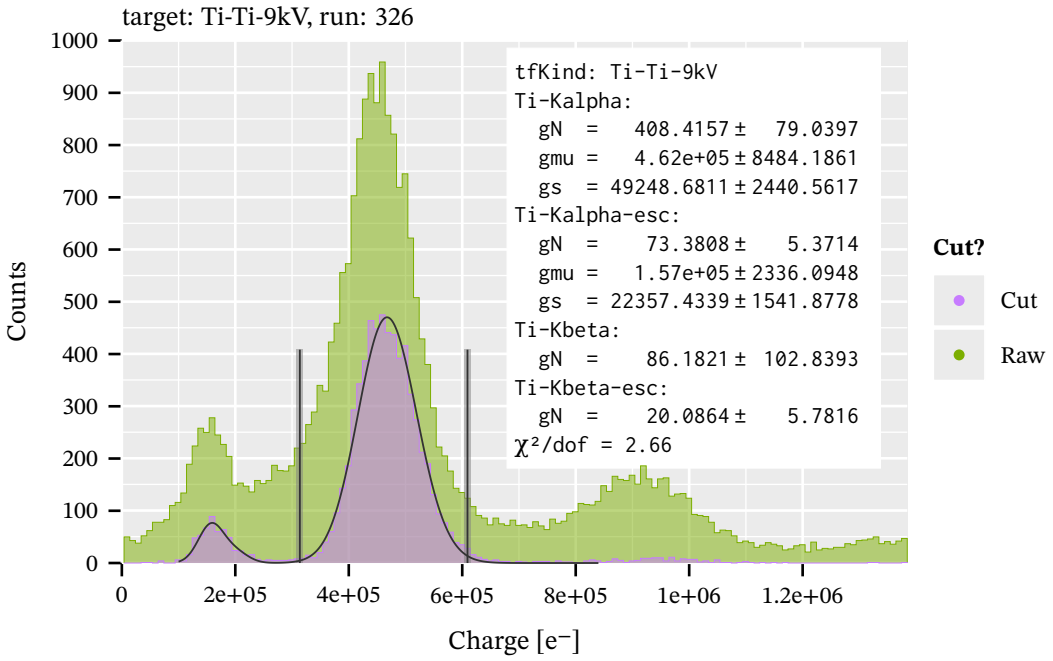


Figure 11.4.: Charge spectrum of the Ti − Ti spectrum at 9 kV from run 326. The green histogram shows the raw data of this run and the purple histogram indicating the data left after the cleaning cuts are applied. The purple line indicates the result of the fit as described in tab. 11.5 with the resulting parameters shown in the box. The black lines represent the 3σ region around the main fluorescence line (with grey error bands), which is later used to extract those clusters likely from the fluorescence line and therefore known energy.

11.2.4. Overview of CDL data in energy

With the fits to the charge spectra performed on a run-by-run basis, they can be utilized to calibrate the energy of each cluster in the data. [6] This is done by using the linear relationship between charge and energy and therefore using the charge of the main fluorescence line as computed from the fit. Each run is therefore self-calibrated (in contrast to our normal energy calibration approach 10.1). Fig. 11.5 shows normalized histograms of all CDL data after applying basic cuts and performing said energy calibrations.

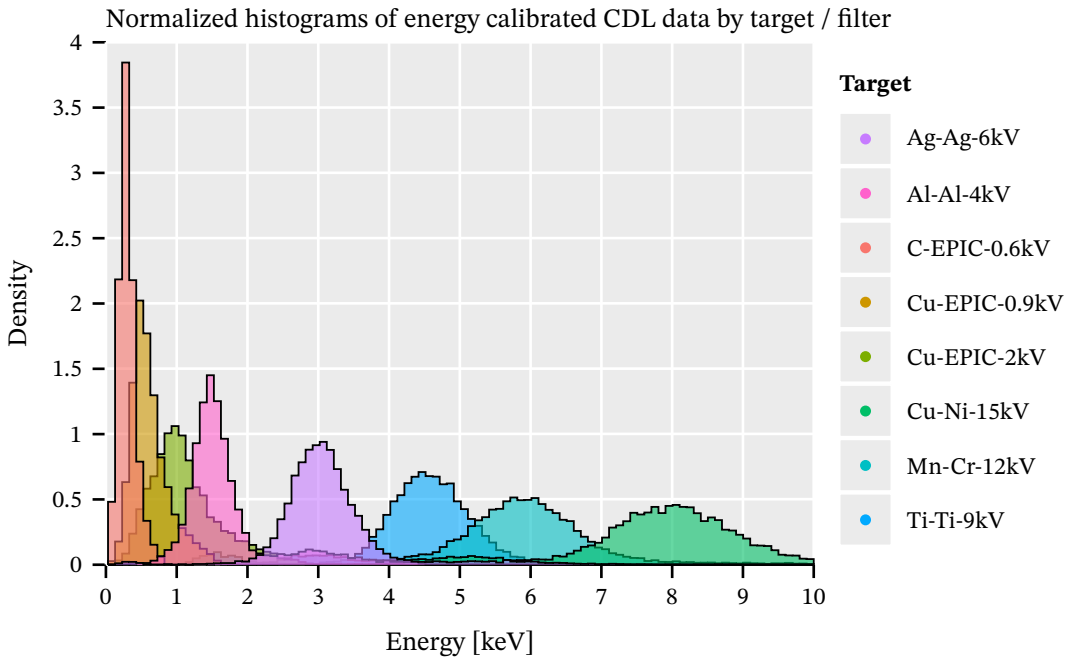


Figure 11.5.: Normalized histograms of all CDL data after applying basic cuts and calibrating the data in energy using the charge of the main fitted line and its known energy as a baseline. Some targets show a wider distribution, due to detector variability which results in different gas gains and thus different charges in different runs.

11.2.5. Definition of the reference distributions

Having performed fits to all charge spectra of each run and the position of the main fluorescence line and its width determined, the reference distributions for the cluster properties entering the likelihood can be computed. By taking all clusters within the 3σ bounds around the main fluorescence line of the data used for the fit, the dataset is selected. This guarantees to leave mainly X-rays of the targeted

[6] We can of course apply the regular energy calibration based on the multiple fits to the CAST calibration data as explained in sec. 10.3. However, due to the very different gas gains observed in the CDL, the applicability is not ideal. Add to that the fact that we know for certain the energy of the clusters in the main fluorescence peak there is simply no need either.

energy for each line in the dataset. As the fit is performed for each run separately, the 3σ charge cut is performed run by run and then all data combined for each fluorescence line (target, filter & HV setting).

The desired reference distributions then are simply the normalized histograms of the clusters in each of the properties. With 8 targeted fluorescence lines and 3 properties this yields a total of 24 reference distributions. Each histogram is then interpreted as a probability density function (PDF) for clusters 'matching' (more on this in sec. 11.2.6) the energy of its fluorescence line. An overview of all the reference distributions is shown in fig. 11.6. We can see that all distributions tend to get wider towards lower energies (towards the top of the plot). This is expected and due to smaller clusters having less primary electrons and therefore statistical variations in geometric properties playing a more important role. [7] The binning shown in the figure is the exact binning used to define the PDF. In case of the fraction of pixels within a transverse RMS radius, bins with significantly higher counts are observed at low energies. This is not a binning artifact, but a result of the definition of the variable. The property computes the fraction of pixels that lie within a circle of the radius corresponding to the transverse RMS of the cluster around the cluster center (see fig. 8.2). At energies with few pixels in total, the integer nature of N or $N + 1$ primary electrons (active pixels) inside the radius becomes apparent.

The binning in the histograms is chosen by hand such that the binning is as fine as possible without leading to significant statistical fluctuations, as those would have a direct effect on the PDFs leading to unphysical effects on the probabilities. Ideally an approach of either an automatic bin selection algorithm or something like a kernel density estimation should be used. However, the latter is slightly problematic due to the integer effects in the low energies of the fraction in transverse RMS variable.

The summarized 'recipe' of the approach is therefore:

1. apply cleaning cuts according to tab. 11.4,
2. perform fits according to tab. 11.5,
3. cut to the 3σ region around the main fluorescence line of the performed fit (i.e. the first term in tab. 11.5),
4. combine all remaining clusters for the same fluorescence line from each run,
5. compute a histogram for each desired cluster property and each fluorescence line,
6. normalize the histogram to define the reference distribution, $\mathcal{P}_i$

[7] Keep in mind that all energies undergo roughly the same amount of diffusion (aside from the energy dependent absorption length to a lesser extent), so lower energy events are more sparse.

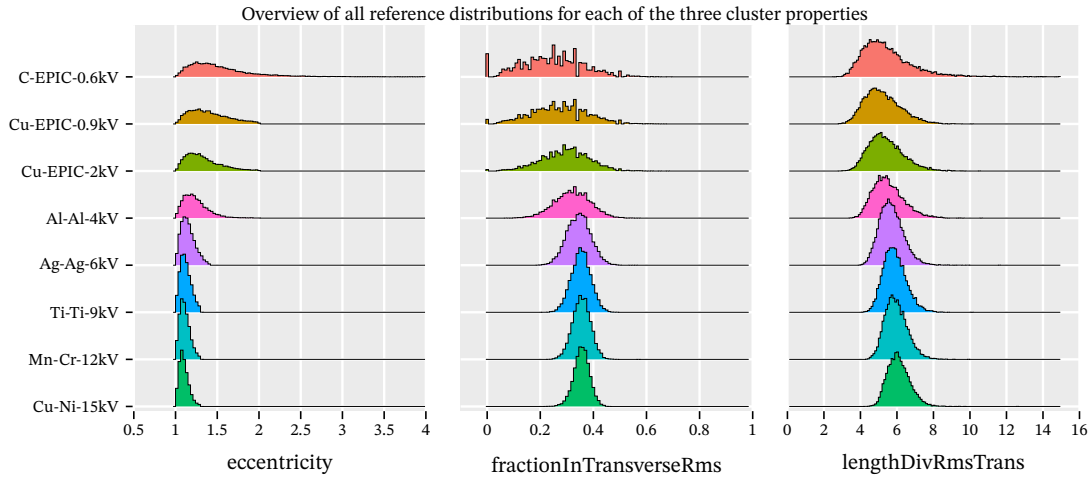


Figure 11.6.: Overview of all reference distributions for each target/filter combination and property. The binning is the same as used to compute the probabilities. Towards lower energies (towards the top) the distributions all become wider, as the clusters have fewer electrons and statistical fluctuations play a larger role. The 'fraction in transverse RMS' property becomes partially discrete, which is not a binning effect, but due to integer counting effects of the number of electrons within the transverse RMS radius.

11.2.6. Definition of the likelihood distribution

With our reference distributions defined it is time to look back at the equation for the definition of the likelihood, eq. 11.1.

> Note: To avoid numerical issues dealing with very small probabilities, the actual relation evaluated numerically is the negative log of the likelihood:

$$-\ln \mathcal{L}(\varepsilon, f, l) = -\ln \mathcal{P}_\varepsilon(\varepsilon) - \ln \mathcal{P}_f(f) - \ln \mathcal{P}_l(l).$$

By considering a single fluorescence line, the three reference distributions $\mathcal{P}_i(i)$ make up the likelihood $\mathcal{L}(\varepsilon, l, f)$ *function* for that energy; a function of the variables ε, f, l. In order to use the likelihood function as a classifier for X-ray like clusters we need a one dimensional expression. This is where the likelihood *distribution* $\mathfrak{L}$ comes in. We reuse all the clusters used to define the reference distributions $\mathcal{P}_{\varepsilon,f,l}$ and compute their likelihood values $\{\mathcal{L}_j\}$, where j is the index of the j-th cluster. By then computing the histogram of the set of all these likelihood values, we obtain the likelihood distribution

$$\mathfrak{L} = \text{histogram}\left(\{\mathcal{L}_j\}\right).$$

All these distributions are shown in fig. 11.7 as negative log likelihood distributions. [8] We see that the distributions change slightly in shape and move towards larger $-\ln \mathcal{L}$ values towards the lower

[8] The $-\ln \mathcal{L}$ values in the distributions give a good idea of why. Roughly speaking the values go from 5 to 20, meaning the actual likelihood values are in the range from $e^{-5} \approx 7 \times 10^{-3}$ to $e^{-20} \approx 2 \times 10^{-9}$! While 64-bit floating point numbers

energy target/filter combinations. The shape change is most notably due to the significant integer nature of the 'fraction in transverse RMS' f variable, as seen in fig. 11.6. The shift to larger values expresses that the reference distributions $\mathcal{P}_i$ become wider and thus each bin has lower probability.

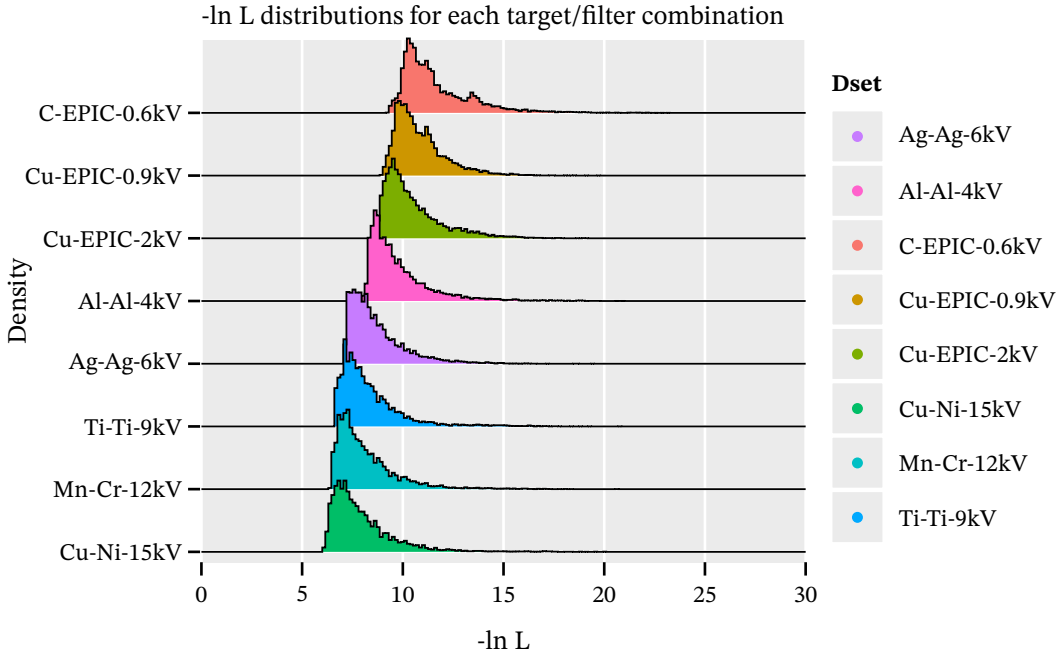


Figure 11.7.: $-\ln\mathcal{L}$ distributions for each of the targeted fluorescence lines and thus target/filter combinations.

To finally classify events as signal or background using the likelihood distribution, one sets a desired "software efficiency" ε_{eff}, which is defined as:

$$\varepsilon_{\text{eff}} = \frac{\int_0^{\mathcal{L}'} \mathfrak{L}(\mathcal{L})\,\mathrm{d}\mathcal{L}}{\int_0^{\infty} \mathfrak{L}(\mathcal{L})\,\mathrm{d}\mathcal{L}}. \tag{11.3}$$

The likelihood *value* $\mathcal{L}'$ is the value corresponding to the $\varepsilon_{\text{eff}}^{\text{th}}$ percentile of the likelihood *distribution* $\mathfrak{L}$. In practical terms one computes the normalized cumulative sum of the log likelihood and searches for the point at which the desired ε_{eff} is reached. The typical software efficiency we aim for is 80 %. Classification as X-ray-like then is simply any cluster with a $-\ln\mathcal{L}$ value smaller than the cut value $-\ln\mathcal{L}'$. Note that this value $\mathcal{L}'$ has to be determined for each likelihood distribution.

To summarize the derivation of the likelihood distribution $\mathfrak{L}$ and its usage as a classifier as a 'recipe':

1. compute the reference distributions $\mathcal{P}_i$ as described in sec. 11.2.5,
2. take the raw cluster data (unbinned data!) of those clusters that define the $\mathcal{P}_i$ and feed each of these into eq. 11.1 for a single likelihood *value* $\mathcal{L}_i$ each,

nowadays in principle provide enough precision for these numbers, human readability is improved. But 32-bit floats would already accrue serious floating point errors due to only about 7 decimal digits accuracy. But even with 64 bit floats, slight changes to the likelihood definition might run into trouble as well.

3. compute the *histogram* of the set of all these likelihood values $\{\mathcal{L}_i\}$ to define the likelihood *distribution* $\mathfrak{L}$.

4. define a desired 'software efficiency' ε_{eff} and compute the corresponding likelihood *value* $\mathcal{L}_c$ using eq. 11.3,

5. any cluster with $\mathcal{L}_i \leq \mathcal{L}_c$ is considered X-ray-like with efficiency ε_{eff}.

Note that due to the usage of negative log likelihoods the raw data often contains infinities, which are just a side effect of picking up a zero probability from one of the reference distributions for a particular cluster. In reality the reference distributions should be a continuous distribution that is nowhere exactly zero. However, due to limited statistics there is a small range of non-zero probabilities (most bins are empty outside the main range). For all practical purposes this does not matter, but it does explain the rather hard cutoff from 'sensible' likelihood values to infinities in the raw data.

11.2.7. Energy interpolation of likelihood distributions

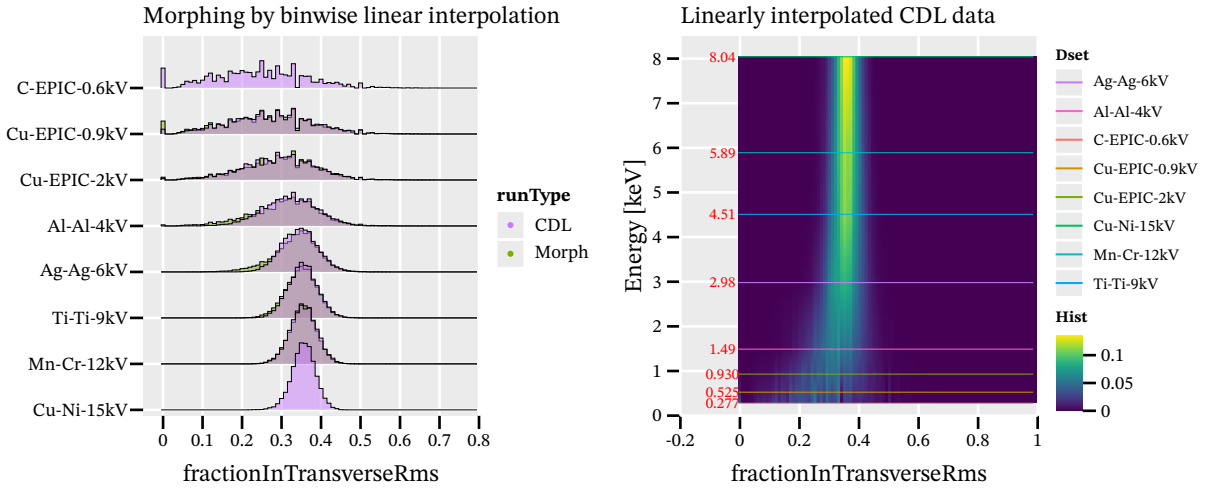
For the GridPix detector used in 2014/15, similar X-ray tube data was taken and each of the 8 X-ray tube energies were assigned an energy interval. The likelihood distribution to use for each cluster was chosen based on which energy interval the cluster's energy falls into. This leads to discontinuities of the properties at the interval boundaries due to the energy dependence of the cluster properties as seen in fig. 11.1(b). This change is of course continuous instead of discrete. It can then lead to jumps in the efficiency of the background suppression method and thus in the achieved background rate. It seems a safe assumption that the reference distributions undergo a continuous change for changing energies of the X-rays. Therefore, to avoid discontinuities we perform a linear interpolation for each cluster with energy E_β between the closest two neighboring X-ray tube energies E_α and E_γ in each probability density $\mathcal{P}_i$ at the cluster's properties. With $\Delta E = E_\alpha - E_\gamma$ the difference in energy between the closest two X-ray tube energies, each probability density is then interpolated to:

$$\mathcal{P}_i(E_\beta, x_i) = \left(1 - \frac{|E_\beta - E_\alpha|}{\Delta E}\right) \cdot \mathcal{P}_i(E_\alpha, x_i) + \left(1 - \frac{|E_\gamma - E_\beta|}{\Delta E}\right) \cdot \mathcal{P}_i(E_\gamma, x_i).$$

Each probability density of the closest neighbors is evaluated at the cluster's property x_i and the linear interpolation weighted by the distance to each energy is computed.

The choice for a linear interpolation was made after different ideas were tried. Most importantly, a linear interpolation does not yield unphysical results (for example negative bin counts in interpolated data, which can happen in a spline interpolation) and yields very good results in the cases that can be tested, namely reconstructing a known likelihood distribution B by doing a linear interpolation between the two outer neighbors A and C. 11.8(a) shows this idea using the fraction in transverse RMS variable. This variable is shown here as it has the strongest obvious shape differences going from

line to line. The green histogram corresponds to interpolated bins based on the difference in energy between the line above and below the target (hence it is not done for the outer lines, as there is no 'partner' above / below). Despite the interpolation covering an energy range almost twice as large as used in practice, over- or underestimation of the interpolation (green) is minimal. To give an idea of what this results in for the interpolation, fig. 11.8(b) shows a heatmap of the same variable showing how the interpolation describes the energy and fraction in transverse RMS space continuously.



(a) Known lines

(b) All energies

Figure 11.8.: (a) shows recovering the known lines (aside from the outer ones) using a binwise linear interpolation from the neighbors. While the interpolation (green) sometimes over- or undershoots it needs to be kept in mind it covers almost twice the energy that is needed. (b) is a heatmap of the full energy vs. fraction in transverse RMS space interpolated using this binwise linear interpolation. The energy of the fluorescence lines is indicated on which the interpolation is based.

Given that such an interpolation works as well as it does on recovering a known line, implies that a linear interpolation on 'half' [9] the energy interval in practice should yield reasonably realistic distributions, certainly much better than allowing a discrete jump at specific boundaries.

See appendix I for a lot more information about the ideas considered and comparisons to the approach without interpolation. In particular fig. I.1, which computes the $\ln \mathcal{L}$ values for all of the (cleaned, cuts tab. 11.4 applied) CDL data comparing the case of no interpolation with interpolation showing a clear improvement in the smoothness of the point cloud.

11.2.8. Energy resolution

On a slight tangent, with the main fluorescence lines fitted in all the charge spectra, the position and line width can be used to compute the energy resolution of the detector:

[9] It's not half the energy as the lines are not evenly spaced in energy.

$$\Delta E = \frac{\sigma}{\mu}$$

where σ is a measure of the line width and μ the position (in this case in total cluster charge). Note that in some cases the full width at half maximum (FWHM) is used and in others the standard deviation (for a normal distribution the FWHM is about 2.35σ). The energy resolutions obtained by our detector in the CDL dataset is seen in fig. 11.9. At the lowest energies below 1 keV the resolution is about $\Delta E \approx 30\,\%$ and generally between 10 to 15 % from 2 keV on.

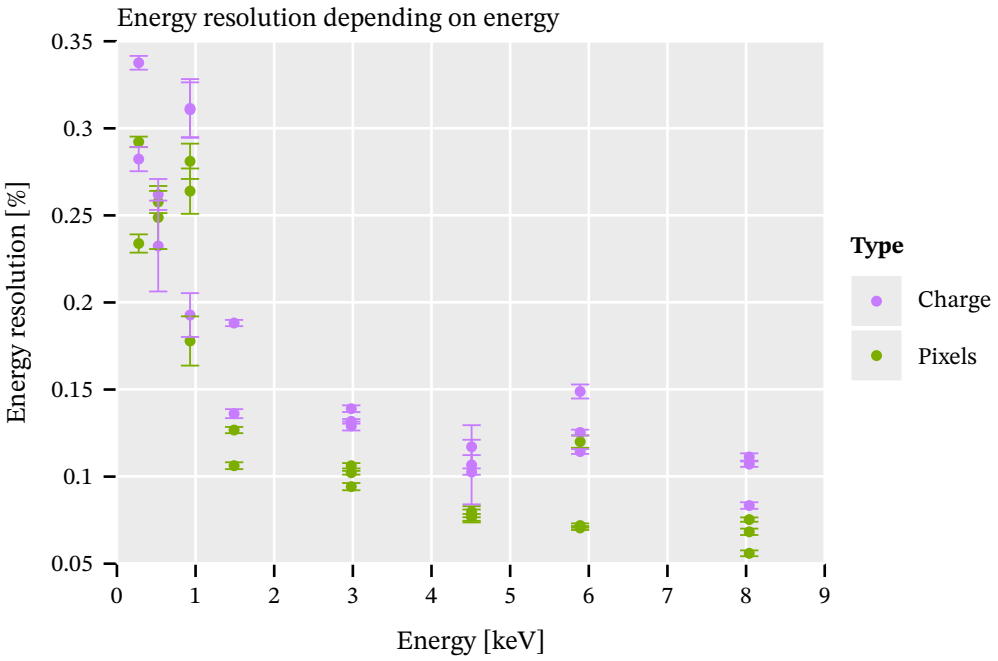


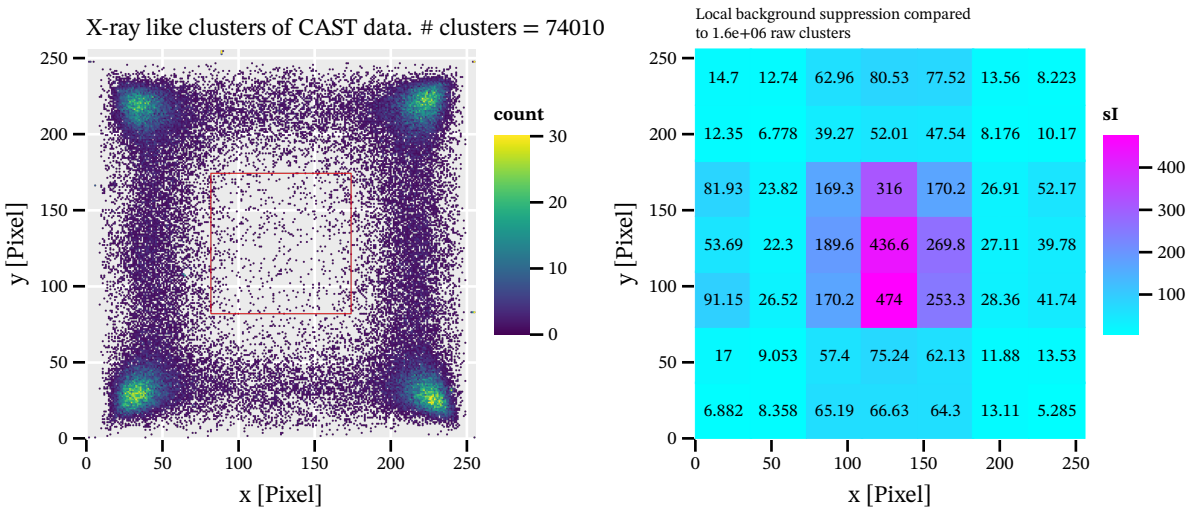
Figure 11.9.: Energy resolutions depending on the energy of the fluorescence lines based on the charge and pixel spectra. As expected the behavior of the energy resolution is more or less $1/E$. The uncertainty for each point is the error propagated uncertainty based on the fit parameter uncertainties for the mean position and the line width. There are multiple data points for each energy owing to the fact that each run is fit separately.

11.3. Application of likelihood cut for background rate

By applying the likelihood cut method introduced in the first part of this chapter to the background data of CAST, we can extract all clusters that are X-ray-like and therefore describe the irreducible background rate. Unless otherwise stated the following plots use a software efficiency of 80 %.

Fig. 11.10(a) shows the cluster centers and their distribution over the whole center GridPix, which highlights the extremely uneven distribution of background. The increase towards the edges and in particular the corners is due to events being cut off. Statistically by cutting off a piece of a track-like event, the resulting event likely becomes more spherical than before. In particular in a corner where

two sides are cut off potentially (see also sec. 6.10). This is an aspect the detector vetoes help with, see sec. 11.5.3. In addition the plot shows some smaller regions of few pixel diameter that have more activity, due to minor noise. With about 74 000 clusters left on the center chip, the likelihood cut at 80 % software efficiency represents a background suppression of about a factor 20 (compare tab. 9.4, $\sim 1.5 \times 10^6$ events on the center chip over the entire CAST data taking). In the regions towards the center of the chip, the suppression is of course much higher. Fig. 11.10(b) shows what the background suppression looks like locally when comparing the number of clusters in a small region of the chip to the total number of raw clusters that were detected. Note that this is based on the assumption that the raw data is homogeneously distributed. See appendix J for occupancy maps of the raw data.



(a) Cluster centers ln $\mathcal{L}$

(b) Background suppression

Figure 11.10.: (a) Cluster centers of all X-ray like clusters in the 2017/18 CAST background data. The number of these clusters increases drastically towards the edges and in particular corners, due to geometric effects. Some regions with minor sparking at the edges are visible as small yellow points of few pixel sizes. The red outline is the center 5 × 5 mm² region in which we quote the background rate. (b) shows the local background suppression over the total number of raw clusters detected. It assumes a homogeneous background distribution in the raw data.

The distribution of the X-ray like clusters in the background data motivate on the one hand to consider local background rates for a physics analysis and at the same time the selection of a specific region in which a benchmark background rate can be defined. For this purpose [134] defines different detector regions in which the background rate is computed and treated as constant. One of these, termed the 'gold region' is a square around the center of 5 mm side length (visible as red square in fig. 11.10(a) and a bit less than 3x3 tiles around center in fig. 11.10(b)). All background rate plots unless otherwise specified in the remainder of the thesis always refer to this region of low background.

Using the $\ln \mathcal{L}$ approach for the GridPix data taken at CAST in 2017/18 at a software efficiency of $\varepsilon_{\text{eff}} = 80\,\%$ then, a background rate shown in fig. 11.11 is achieved. The average background rate between 0 to 8 keV in this case is $2.124(97) \times 10^{-5}\,\text{keV}^{-1}\,\text{cm}^{-2}\,\text{s}^{-1}$.

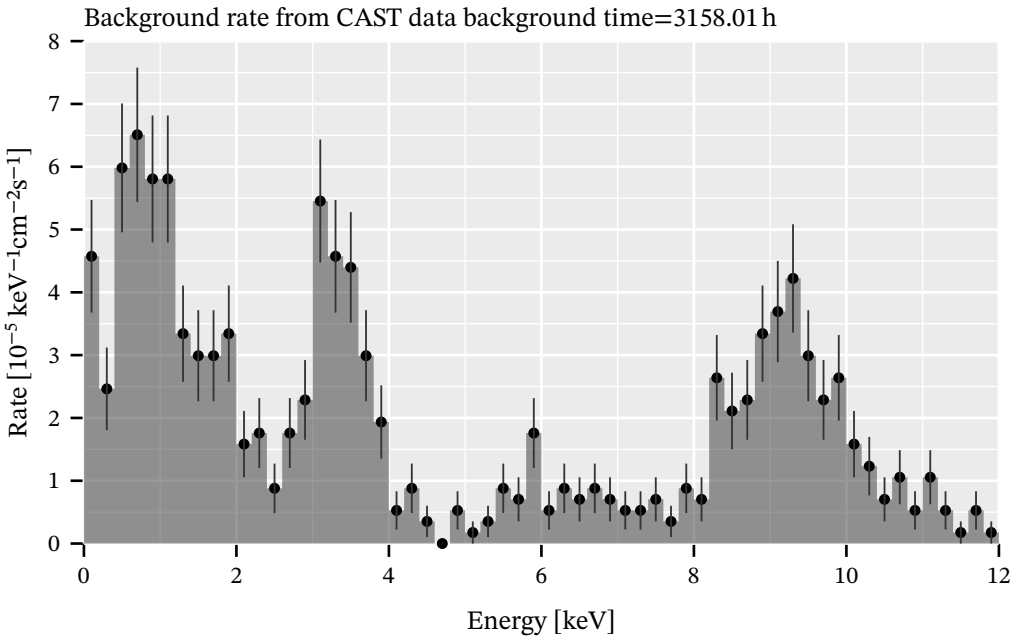


Figure 11.11.: Background rate achieved based on the $\ln \mathcal{L}$ method at $\varepsilon_{\text{eff}} = 80\,\%$ using the CAST 2017/18 data without the application of any vetoes.

This is comparable to the background rate presented in [134], and reproduced here in in fig. 6.4 for the 2014/15 CAST GridPix detector. For this result no additional detector features over those available for the 2014/15 detector are used and the classification technique is essentially the same. [10] In order to improve on this background rate we will first look at a different classification technique based on artificial neural networks. Then afterwards we will go through the different detector features and discuss how they can further improve the background rate.

11.4. Artificial neural networks as cluster classifiers

The likelihood cut based method presented in section 12.2 works well, but does not use the full potential of the data. It mainly uses length and eccentricity related properties, which are hand picked and ignores the possible separation power of other properties.

Multiple different ways to use all separation power exist. One promising approach is the usage of artificial neural networks (ANNs). A multi-layer perceptron (MLP) [7] [11] is a simple supervised ANN

[10] The only conceptual difference in the techniques is our inclusion of the interpolation between reference distributions in energy.

[11] Due to the rich and long history of artificial neural networks picking "a" or only a few citations is tricky. Amari's work

model, which consists of an input and output layer plus one or more fully connected hidden layers. By training such a network on the already computed geometric properties of the clusters, computational requirements remain relatively moderate compared to approaches using – for example – the raw data as inputs.

As each neuron on a given layer in an MLP is fully connected to all neurons on the previous layer, the output of neuron k on layer i is described by

$$y_{k,i} = \varphi\left(\sum_{j=0}^{m} w_{kj} y_{j,i-1}\right) \tag{11.4}$$

where w_{kj} is the weight between neuron k on layer i and neuron j of m neurons on layer $i-1$. φ is a (typically non-linear) activation function which is applied to saturate a neuron's output.

If $y_{j,i-1}$ is considered a vector for all j, w_{kj} can be considered a weight matrix and eq. 11.4 is simply a matrix product. Each layer is computed iteratively starting from the input layer towards the output layer.

Given that an MLP is a supervised learning algorithm, the desired target output for a given input during training is known. A loss function is defined to evaluate the accuracy of the network. Many different loss functions are used in practice, but in many cases the mean squared error (MSE; also known as L2 norm) is used as a loss

$$l(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

where $\mathbf{y}$ is a vector $\in \mathbb{R}^N$ of the network outputs and $\hat{\mathbf{y}}$ the target outputs. The sum runs over all N output neurons. [12]

In order to train a neural network, the initially random weights must be modified. This is done by computing the gradients of the loss (given an input) with respect to all the weights of the network, $\frac{\partial l(\mathbf{y})}{\partial w_{ij}}$. Effectively the chain rule is used to express these partial derivatives using the intermediate steps of the calculation. This leads to an iterative equation for the weights further up the network (towards the input layer). Each weight is updated from iteration n to $n+1$ according to

$$w_{ij}^{n+1} = w_{ij}^{n} - \eta \frac{\partial l(\mathbf{y})}{\partial w_{ij}^{n}}.$$

[7] was, as far as I'm aware, the first to combine a perceptron with non-linear activation functions and using gradient descent for training. See Schmidhuber's recent overview for a detailed history leading up to modern deep learning [190].

[12] For performance reasons to utilize the parallel nature of GPUs, training and inference of NNs is done in 'mini-batches'. As still only a single loss value is needed for training, the loss is computed as the mean of all losses for each mini-batch element.

where η is the learning rate. This approach to updating the weights during training is referred to as backpropagation. [184] [13]

11.4.1. MLP for CAST data

The simplest approach to using a neural network for data classification of CAST-like data, is an MLP that uses the pre-computed geometric properties for each cluster as an input.

The choice remains between a single or two output neurons. The more classical approach is treating signal (X-rays) and background events as two different classes that the classifier learns to predict, which we also use. By our convention the target outputs for signal and background events are

$$\hat{\mathbf{y}}_{\text{signal}} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$
$$\hat{\mathbf{y}}_{\text{background}} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

where the first entry of each $\hat{\mathbf{y}}$ corresponds to output neuron 1 and the second to output neuron 2.

For the network to generalize to all real CAST data, the training dataset must be representative of the wide variety of the real data. For background-like clusters it can be sourced from the extensive non-tracking dataset recorded at CAST. For the signal-like X-ray data it is more problematic. The only source of X-rays with enough statistics are the ^{55}Fe calibration datasets from CAST, but these only describe X-rays of two energies. The CAST detector lab data from the X-ray tube are both limited in statistics as well as suffering from systematic differences to the CAST data due to different gas gains. [14]

We will now describe how we generate X-ray clusters for MLP training by simulating them from a *target energy*, a specific *transverse diffusion* and a desired *gas gain*.

[13] The gradients in a neural network are usually computed using 'automatic differentiation' (or 'autograd', 'autodiff'). There are two forms of automatic differentiation: forward mode and reverse mode. These differ by the practical evaluation order of the chain rule. Forward mode computes the chain rule from left-to-right (input to output), while reverse mode computes it from right-to-left (output to input). Computationally these differ in their complexity in terms of the required number of evaluations given N inputs and M outputs. Forward mode computes all M output derivatives for a single input, whereas reverse mode computes all input derivatives for a single output. Thus, forward mode is efficient for cases with few inputs and many outputs, while the opposite is true for reverse mode. Neural networks are classical cases of many inputs to few outputs (scalar loss function!). As such, reverse mode autograd is the standard way to compute the gradients during NN training. In the context it is effectively synonymous with 'backpropagation' (due to its output-to-input evaluation of the chain rule).

[14] The acute reader may wonder why we care less about this for the $\ln \mathcal{L}$ method. The reason is simply that the effect of gas gain on the three properties used there is comparatively small. But if (almost) all properties are to be used, that is less true.

11.4.2. Generation of simulated X-rays as MLP training input

To generate simulated events we wish to use the least amount of inputs, which still yield events that represent the recorded data as well as possible. In particular, the systematic variations between different times and datasets should be reproducible based on these parameters. The idea is to generate events using the underlying gaseous detector physics (see sec. 5) and make as few heuristic modifications to better match the observed (imperfect) data as possible. This lead to an algorithm, which only uses three parameters: a target energy for the event (within the typical energy resolution). A gas gain to encode the gain variation seen. A gas diffusion coefficient to encode variations in the diffusion properties (also possibly a result of changing temperatures).

In contrast to approaches that simulate the interactions at the particle level, our simulation is based entirely on the emergent properties seen as a result of those. The basic Monte Carlo algorithm will now be described in a series of steps.

1. (optional) sample from different fluorescence lines for an element given their relative intensities to define a target energy E.

2. sample from the exponential distribution of the absorption length (see sec. 5.1.1) for the used gas mixture and target photon energy to get the conversion point of the X-ray in the gas (Note: we only sample X-rays that convert; those that would traverse the whole chamber without conversion are ignored).

3. sample a target charge to achieve for the cluster, based on target energy by sampling from a normal distribution with a mean roughly matching detector energy resolution (target energy inverted to a charge).

4. sample center position of the cluster within a uniform radius of 4.5 mm around the chip center.

5. begin the sampling of each electron in the cluster.

6. sample a charge (in number of electrons after amplification) for the electron based on a Pòlya distribution of input gas gain (and matching normalization & θ constant). Reject the electron if not crossing activation threshold (based on real data).

7. sample a radial distance from the cluster center based on gas diffusion constant D_T and diffusion after the remaining drift distance to the readout plane z_{drift} using eq. 5.9. Sample twice from the 1D version using a normal distribution $\mathcal{N}(\mu = 0, \sigma = D_T \cdot \sqrt{z_{\text{drift}}})$ for each dimension. Combine to a radius from center, $r = \sqrt{x^2 + y^2}$.

8. sample a random angle, uniform in $(0, 2\pi]$.

9. convert radius and angle to an (x, y) position of the electron, add to the cluster.

10. based on a linear approximation activate between 0 to 4 neighboring pixels with a slightly reduced gas gain. We never activate any neighbors at a charge of 1000 e^- and always activate at least one at 10 000 e^-. Number of activated pixels depends on a uniform random number being below one of 4 different thresholds:

$$N_{\text{neighbor}} = \text{rand}(0, 1) \cdot N < \text{threshold}$$

    where the threshold is determined by the linear function described by the mentioned condition and rand$(0, 1)$ is a uniform random number in $(0, 1)$.

11. continue sampling electrons until the total charge adds up to the target charge. We stop at the value closest to the target (before or after adding a final pixel that crosses the target charge) to avoid biasing us towards values always larger than the target.

12. The final cluster is reconstructed just like any other real cluster, using the same calibration functions as the real chip (depending on which dataset it should correspond to).

Note: Neighboring pixels are added to achieve matching eccentricity distributions between real data and simulated. Activating neighboring pixels increases the local pixel density randomly, which effectively increases the weight of some pixels, leading to a slight increase of eccentricity of a cluster. The approach of activating up to 4 neighbor pixels and giving them slightly lower charges stems from empirically matching the simulated data to real data. From a physical perspective the most likely cause of neighboring pixels is due to UV photons that are emitted in the amplification region and travel towards the grid and produce a new electron, which starts an avalanche from there. From that point of view neighboring pixels should see the full gas amplification and neighbors other than the direct { up, down, left, right } neighbors can be activated (depending on an exponential distribution due to possible absorption of the UV photons in the gas). See Markus Gruber's master thesis [96] for a related study on this for GridPix detectors.

Based on the above algorithm, fake events can be generated that either match the gas gain and gas diffusion of an existing data taking run (background or calibration) or any arbitrary combination of parameters. The former is important for verification of the validity of the generated events as well as to check the MLP cut efficiency (more on that in sec. 11.4.6). The latter is used to generate a wide variety of MLP training data. Event production is a very fast process [15], allowing to produce large amounts of statistics for MLP training in a reasonable time frame.

[15] Generating 100 000 fake events of the ^{55}Fe photopeak takes around 35 s on a single thread using an AMD Ryzen 9 5950X. The code is not exactly optimized either.

For other applications that require higher fidelity simulations, Degrad [41], a sister program to Magboltz [42], can be used, which simulates the particle interactions directly. This is utilized in a code by M. Gruber [95], which uses Degrad and Garfield++ [89] to perform realistic event simulation for GridPix detectors (its data can also be analyzed by the software presented in sec. 8.1 and appendix O).

11.4.3. Determination of gas diffusion

To generate X-ray events with similar physical properties as those seen in a particular data taking run, we need the three inputs required: gas gain, gas diffusion and target energy. The determination of the gas gain is a well defined procedure, explained in sections 7.1.2 and 10.2.2. The target energy is a matter of the purpose the generated data is for. The gas diffusion however is more complicated and requires explanation.

In theory the gas diffusion is a fixed parameter for a specific gas mixture at fixed temperature, pressure and electromagnetic fields and can be computed using Monte Carlo tools as mentioned already in sec. 5.3.4. However, in practice MC tools suffer from significant uncertainty (in particular in the form of run-by-run RNG variation) especially at common numbers of MC samples. More importantly though, real detectors do not have perfectly stable and known temperatures, pressures and electromagnetic fields (for example the field inhomogeneity of the drift field in the 7-GridPix detector is not perfectly known), leading to important differences and variations from theoretical numbers.

Fortunately, similar to the gas gain, the effective numbers for the gas diffusion can be extracted from real data. The gas diffusion constant $D_t(z)$ (introduced in sec. 5.3.4) describes the standard deviation of the transverse distance from the initial center after drifting a distance z along a homogeneous electric field. This parameter is one of our geometric properties, in the form of the transverse RMS 'RMS$_T$ ' (which is actually the transverse standard deviation of the electron positions [16]) and thus it can be used to determine the gas diffusion coefficient. [17] As it is a statistical process and D_t is the standard deviation of the population a large ensemble of events is needed.

This is possible both for calibration data as well as background data. In both cases it is important to apply minor cuts to remove the majority of events in which RMS$_T$ contains contributions that are not due to gas diffusion (e.g. a double X-ray event which was not separated by the cluster finder). For

[16] σ_T would be more appropriate following eq. 5.9 in sec. 5.3.4.

[17] Note that it is important to consider the transverse standard deviation and not the longitudinal one. Even for an X-ray there will be a short track like signature during production of the primary electrons from the initial photoelectron. This can define the long axis of the cluster in certain cases (depending on gas mixture). Similarly, for background events of cosmic muons the long axis corresponds to the direction of travel while the transverse direction is purely due to the diffusion process.

calibration data the cuts should filter to single X-rays, whereas for background events clean muon tracks are the target.

In theory one could determine it by way of a fit to the RMS_T distribution of all clusters in a data taking run as was done in [133]. This approach is problematic in practice due to the large variation in possible X-ray conversion points – and therefore drift distances – as a function of energy. Different energies lead to different convolutions of exponential distributions with the gas diffusion. This means a single fit does not describe the data well in general and the upper limit is not a good estimator for the real gas diffusion coefficient (because it is due to those X-rays converting directly behind the detector window by chance and/or undergoing statistically large amount of diffusion).

Instead of performing an analytical convolution of the exponential distribution and gas diffusion distributions to determine the correct fit, a Monte Carlo approach is used here as well. The idea here is to use a reasonable [18] gas diffusion constant to generate (simplified [19]) fake events. Compute the RMS_T distribution of this fake data and compare to the real RMS_T distribution of the target run based on a test statistic. Next one computes the derivative of the test statistic as a form of loss function and adjusts the diffusion coefficient accordingly. Over a number of iterations the distribution of the simulated RMS_T distribution will converge to the real RMS_T distribution, if the choice of test statistic is suitable. In other words we use gradient descent to find that diffusion constant D_T, which best reproduces the RMS_T distribution of our target run.

One important point with regards to computing it for background data is the following: when generating fake events for the background dataset, the equivalent 'conversion point' for muons is a uniform distribution over all distances from the detector window to the readout. For X-rays it is given by the energy dependent attenuation length in the detector gas.

The test statistic chosen in practice for this purpose is the Cramér-von-Mises (CvM) criterion, defined by [68, 154]:

$$\omega^2 = \int_{-\infty}^{\infty} \left[F_n(x) - F^*(x)\right]^2 \, \mathrm{d}F^*(x)$$

where $F_n(x)$ is an empirical distribution function to test for and $F^*(x)$ a cumulative distribution function to compare with. In case of the two-sample case it can be computed as follows [10]:

$$T = \frac{NM}{N+M}\omega^2 = \frac{U}{NM(N+M)} - \frac{4MN-1}{6(M+N)}$$

[18] For example a number provided by Magboltz for the used gas mixture at normal temperature and used chamber pressure. For the CAST gas mixture and settings we start with $D_T = 660\,\mu\text{m}\,\text{cm}^{-1/2}$.

[19] In principle the event generation is the same algorithm as explained previously, but it does not sample from a Pólya for the gas gain, nor compute any cluster properties. Only the conversion point, number of target electrons based on target energy and their positions is simulated. From these electron positions the long and short axes are determined and σ_T returned.

with U:

$$U = N \sum_{i=1}^{N} (r_i - i)^2 + M \sum_{j=1}^{M} \left(s_j - j\right)^2$$

In contrast to – for example – the Kolmogorov-Smirnov test, it includes the entire (E)CDF into the statistic instead of just the largest deviation, which is a useful property to protect against outliers.

The iterative optimization process is therefore

$$D_{T,i+1} = D_{T,i} - \eta \frac{\partial f(D_T)}{\partial D_T}$$

where $f(D_T)$ is the Monte Carlo algorithm, which computes the test statistic for a given D_T and η is the step size along the gradient (a 'learning rate'). The derivative is computed using finite differences. [20]

Fig. 11.12 shows the values of the transverse gas diffusion constant D_T as they have been determined for all runs. The color scale is the value of the Cramér-von-Mises test criterion and indicates a measure of the uncertainty of the obtained parameter. We can see that for the CDL datasets (runs above 305) the uncertainty is larger. This is because reproducing the RMS_T distribution for these datasets is more problematic than for the CAST ^{55}Fe datasets (due to more data impurity caused by X-ray backgrounds of energies other than the target fluorescence line). Generally we can see variation in the range from about 620 to 660 $\mu\text{m}\,\text{cm}^{-1/2}$ for the CAST data and larger variation for the CDL data. The theoretical value we expect is about 670 $\mu\text{m}\,\text{cm}^{-1/2}$, implying the approach yields reasonable values.

Because this iterative calculation is not computationally negligible, the resulting D_T parameters for each run are cached in an HDF5 file. [21]

11.4.4. Comparison of simulated events and real data

Fig. 11.13 shows the comparison of all cluster properties of simulated and real data for one ^{55}Fe calibration run, number 241. The data is cut to the photopeak and normalized to fit into a single plot. Each ridge shows a kernel density estimation (KDE) of the data. Gas gain and gas diffusion are extracted from the data as explained in previous sections.

All distributions outside the number of hits agree extremely well. This is expected to not perfectly match. Only the total charge is used as an input to the MLP and it matches pretty well on its own. The

[20] Initially I tried to compute the gradient using automatic differentiation using dual numbers for the D_T input and the MC code, but the calculation of the test statistics is effectively independent of the input numbers due to the calculation of the ECDF. As such the derivative information is lost. I didn't manage to come up with a way to compute Cramér-von-Mises based on the actual input numbers directly in a timely manner.

[21] One optimization takes on the order of $< 30\,\text{s}$ for most runs. But the diffusion parameters are used many times.

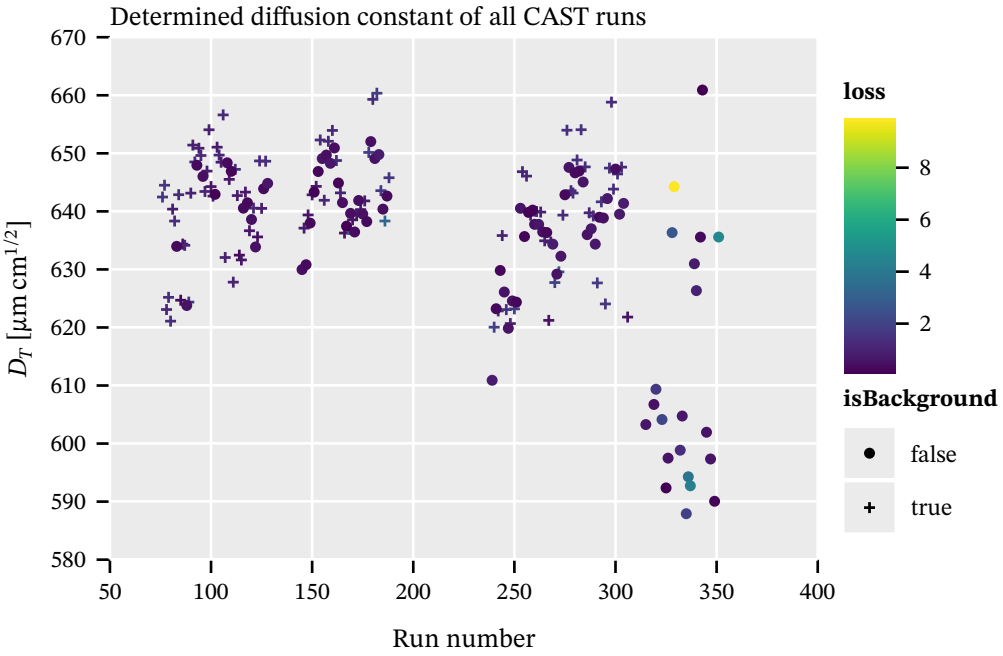


Figure 11.12.: Transverse gas diffusion constant D_T as determined based on an iterative method attempting to reproduce the transverse RMS distribution using simulated X-ray events, the Cramér-von-Mises (CvM) test criterion and gradient descent. All runs above run 305 are CDL calibration runs. The color scale is the CvM test score. Background runs are marked by crosses.

reason is the neighboring activation logic in the MC algorithm, which is empirically modified such that it produces correct geometric behavior at the cost of slightly over- or underestimating the number of hits. The Pólya sampling and general neighboring logic is too simplistic to properly reproduce both aspects at the same time. Other runs show similar agreement. Plots for all runs like this can be found in the extended version of the thesis.

11.4.5. Overview of the best performing MLP

As a reference [22] an MLP was trained on a mix of simulated X-rays and a subset of the CAST background data. The implementation was done using Flambeau [198], a wrapper for libtorch [162] [23], using the parameters and inputs as shown in tab. 11.6. The network layout consists of 14 input neurons, two hidden layers of only 30 neurons [24] each and 2 neurons on the output layer. We use the Adam [129] optimizer and tanh as an activation function on both hidden layers, with sigmoid used

[22] This network is considered a ”reference” implementation, as it is essentially the simplest ANN layout possible.

[23] `libtorch` is the C++ library PyTorch is built on.

[24] Yes, only 30 neurons. When using SGD as an optimizer more neurons are useful. With Adam anything more leads to excessive overtraining. Model performance and training time is anyhow better with Adam. Indeed, even a 10 neuron model performs almost as good. The latter is fun, because it allows to write the weight matrices on a piece of paper and almost do a forward pass by hand.

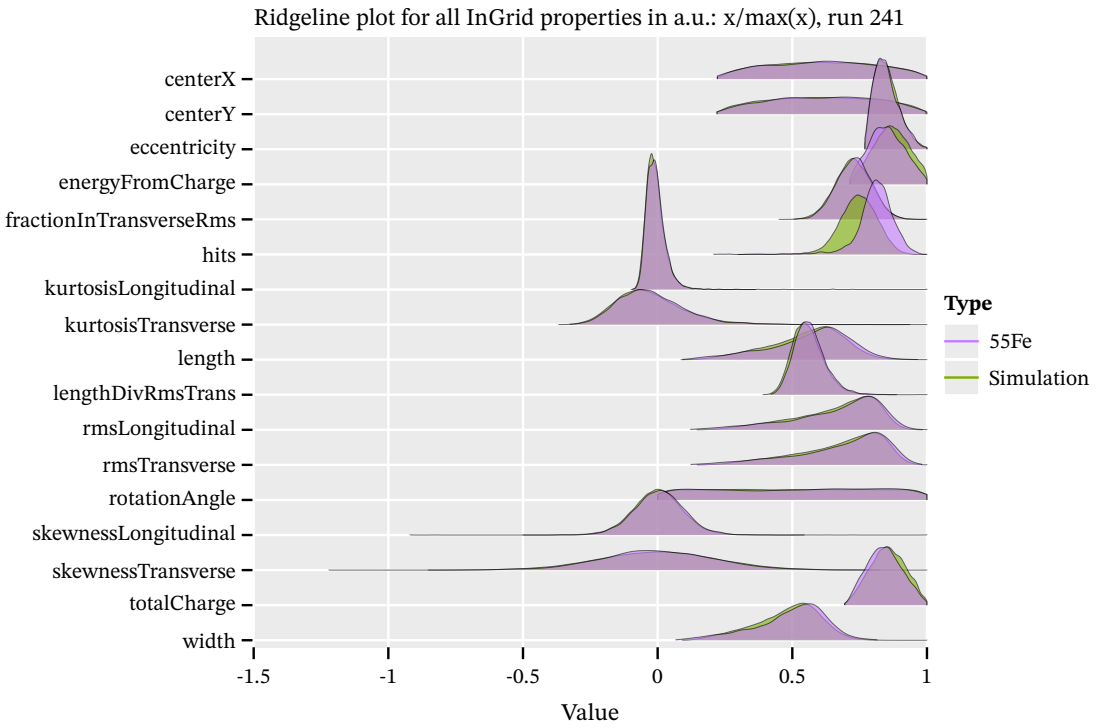


Figure 11.13.: Comparison of all geometric properties of simulated and real data of the ^{55}Fe photopeak clusters of run 241 in a ridgeline plot. Each ridge shows a KDE of the data. The gas gain and gas diffusion were first extracted from the real run data to simulate events of the photopeak. The two datasets agree very well with the exception of the number of hits (expected).

on the output layer. The mean squared error between output and target vectors is used as the loss function.

Property	Value
Input neurons	14 (12 geometric, 2 non-geometric)
Hidden layers	2
Neurons on hidden layer	30
Output neurons	2
Activation function	tanh
Output layer activation function	sigmoid
Loss function	Mean Squared Error (MSE)
Optimizer	Adam [129]
Learning rate	7×10^{-4}
Batch size	8192
Training data	250 000 simulated X-rays and 288 000 real background events selected from the **outer chips only**, same number for validation
Training epochs	82 000

Table 11.6.: Overview of all MLP parameters and training information.

The 14 input neurons were fed with all geometric properties that do not scale directly with the energy (no number of active pixels or direct energy) or position on the chip (as the training data is skewed towards the center) with the exception of the total charge. For the background data we avoid 'contaminating' the training dataset with CAST clusters that will end up as part of our background rate for the limit calculation by only selecting clusters from chips other than the center chip. This way the MLP is trained entirely on data independent of the CAST data of interest. 250 000 synthetic X-rays and 288 000 background clusters are used for training and the same number of validation [25].

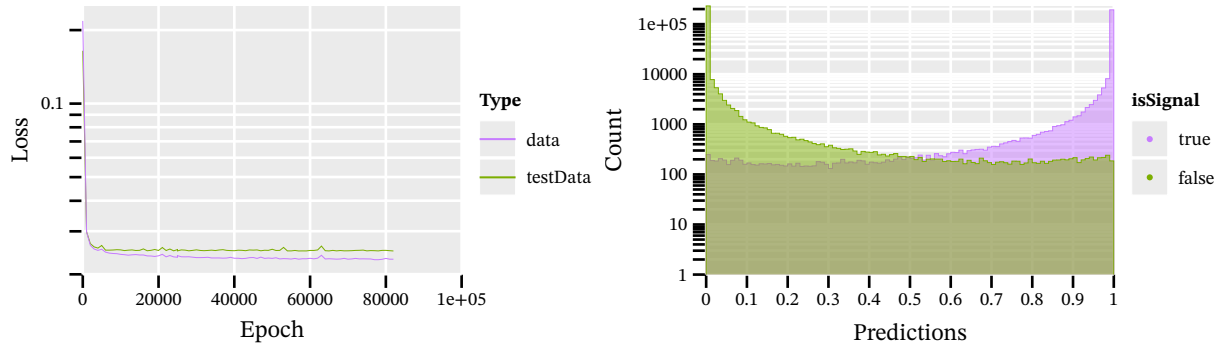
The first 25 000 epochs are trained on a synthetic X-ray dataset with a strong bias to low energy X-rays (clusters in the range from 0 to 3 keV with the frequency dropping linearly with increasing energy). This is because these are more difficult to separate from background. Afterwards we switch to a separate set of synthetic X-rays uniformly distributed in energies. Without this approach the network can drift towards a local minimum, in which low energy X-rays are classified as background, while still achieving good accuracy. At the cost of low energy X-rays, low energy background clusters

[25] For training X-rays a total of 500 000 events are simulated. For background 1000 events are randomly selected from each outer chip for each background runs (so 6000 clusters per background run), totaling 576 000 events. Training and validation is split 50-50.

are almost perfectly rejected. This is of course undesirable.

During the training process of 82 000 epochs, every 1000 epochs the accuracy and loss are evaluated based on the test sample and a model checkpoint is stored. The evolution of the loss function for this network is shown in fig. 11.14(a). We can see that performance for the test dataset is mostly stagnant from around epoch 10 000. The network develops minor overtraining, but this is not important, because no training data is ever used in practice.

Fig. 11.14(b) shows the output of neuron 0 (neuron 1 is a mirror image) for the test sample, with signal like data towards 1 and background like data towards 0. The separation is almost perfect, but a small amount of each type can be seen over the entire range. This is expected due to the presence of real X-rays in the background dataset, low energy events generally being similar and statistical outliers in the X-ray dataset.



(a) Training loss

(b) MLP prediction

Figure 11.14.: (a) Loss over the training progress, evaluated every 1000 epochs. Test data loss mostly stagnant after 10 000 epochs. (b) Output of the validation data sample for neuron 0 as a log plot.

Beyond the MLP prediction to indicate a measure of classification efficiency, we can look at the receiver operating characteristic (ROC) curves for different X-ray reference datasets of the CDL data. For this we read all real center GridPix background data and all CDL data. Both of these are split into the corresponding target/filter combinations based on the energy of each cluster.

We can then define the background rejection b_{rej} for different cut values c_i as

$$b_{\text{rej}} = \frac{\text{card}(\{b < c_i\})}{\text{card}(\{b\})},$$

where b are all the predictions of the MLP for background data and card is the cardinality of the set of numbers, i.e. the number of entries. i refers to the i^{th} cut value (as this is computed with discrete bins). In addition we define the signal efficiency by

$$s_{\text{eff}} = \frac{\text{card}(\{s \geq c_i\})}{\text{card}(\{s\})},$$

where similarly s are the MLP predictions for X-ray clusters.

If we plot the pairs of signal efficiency and background rejection values (one for each cut value), we produce a ROC curve. Done for each CDL target/filter combination and similarly for the likelihood cut by replacing the MLP prediction with the likelihood value of each cluster, we obtain fig. 11.15. The line style indicates the classifier method ($\ln \mathcal{L}$ or MLP) and the color corresponds to the different targets and thus different fluorescence lines. The improvement in background rejection at a fixed signal efficiency exceeds 10 % at multiple energies. The only region where the MLP is worse than the likelihood cut is for the Cu-EPIC 0.9 kV target/filter combination at signal efficiencies above about 92 %. Especially the shape of the ROC curve for the MLP at high signal efficiencies implies that it should produce comparable background rates to the likelihood cut at higher signal efficiencies.

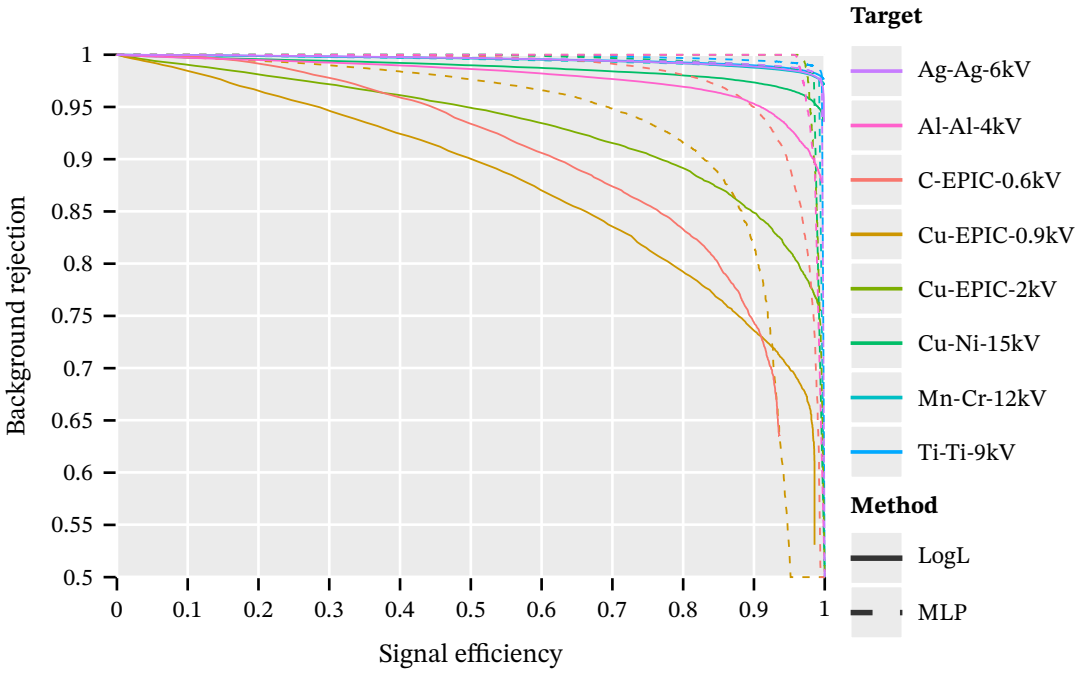


Figure 11.15.: ROC curves for the comparison of the likelihood cut method (solid lines) to the MLP predictions (dashed lines), colored by the different targets used to generate the reference datasets. The background data used for each target corresponds to background clusters in an energy range around the fluorescence line. The MLP visibly outperforms the likelihood cut method in all energies. At the same signal efficiency (x axis) a significantly higher background rejection is achieved.

11.4.6. Determination of MLP cut value

Based on the output distributions of the MLP, fig. 11.14(b), it can be used to act as a cluster discriminator in the same way as for the likelihood cut method, following eq. 11.3. The likelihood distribution is replaced by the MLP prediction via either of the two output neurons. For example, as seen for one output neuron in fig. 11.14(b), the software efficiency would be determined based on the 'prediction'

value along the x-axis such that the desired software efficiency is above the cut value. Any cluster below would be removed. In practice again the empirical distribution function of the signal-like output data is computed and the quantile of the target software efficiency ε_{eff} is determined. Similarly to the likelihood cut method this is done for 8 different energy ranges corresponding to the CDL fluorescence lines. However, no interpolation is performed because only the single output distribution is known. [26]

Due to the significant differences in gas gain between the CDL dataset and the CAST ^{55}Fe calibration data, we do not use a single cut value for each energy for all CAST data taking runs. Instead we use the X-ray cluster simulation machinery to provide run specific X-ray clusters from which to deduce cut values.

For each run we wish to apply the MLP to, we start by computing the mean gas gain based on all gas gain intervals. Then we determine the gas diffusion coefficient as explained in sec. 11.4.3. With two of the three required parameters for X-ray generation in hand, we then simulate X-rays following each X-ray fluorescence line measured in the CDL dataset, yielding 8 different simulated datasets for each run. Based on these we compute one cut value on the MLP output each. The so deduced cut value is applied as the MLP cut to that run in the valid energy range. The same approach is used for calibration runs as well as for background runs.

This means *both* the MLP training as well as determination of cut values is *entirely* based on synthetic data, only using aggregate parameters of the real data.

11.4.7. Verification of software efficiency using calibration data

As the simulated X-ray clusters certainly differ from real X-rays, verification of the software efficiency using calibration data is required. For all ^{55}Fe calibration runs as well as all CDL data we produce cut values as explained in the previous section. Then we apply these to the real clusters of those runs and compute the effective efficiency as

$$\varepsilon_{\text{effective}} = \frac{N_{\text{cut}}}{N_{\text{total}}}$$

where N_{cut} is the clusters remaining after applying the MLP cut. N_{total} is the total number of clusters after application of the standard cleaning cuts applied for the ^{55}Fe spectrum fit and the CDL fluorescence line fits.

The resulting efficiency is the effective efficiency the MLP produces. A close match with the target software efficiency implies the simulated X-ray data matches the real data well and the network learned to identify X-rays based on physical properties (and not due to overtraining). In the limit

[26] One could of course interpolate on the cut values itself, but that is less well motivated and harder to cross-check. Predicting a cut value from two known, neighboring energies would likely not work very well.

calculation later, the mean of all these effective efficiencies of the ^{55}Fe calibration runs is used in place of the target software efficiency as a realistic estimator for the signal efficiency. The standard deviation of all these effective efficiencies is one of the systematics used there.

Fig. 11.16 shows the effective efficiencies obtained for all ^{55}Fe calibration and CDL runs using the MLP introduced earlier. The marker symbol represents the energy of the target fluorescence line. Note that the ^{55}Fe calibration escape peak near 3 keV is a separate data point `"3.0"` compared to the silver fluorescence line given as `"2.98"`. That is because the two are fundamentally different things. The escape event is a 5.9 keV photopeak X-ray entering the detector, which ends up 'losing' about 3 keV due to excitation of an argon fluorescence Kα X-ray. This means the absorption length is that of a 5.9 keV photon, whereas the silver fluorescence line corresponds to a real 2.98 keV photon and thus corresponding absorption length. As such, the geometric properties on average are different.

The target efficiency in the plot was 95 % with an achieved mean effective efficiency close to the target. Values are slightly lower in Run-2 (runs < 200) and slightly above in Run-3. Variation is significantly larger in the CDL runs (runs above number 305), but often towards larger efficiencies. Only one CDL run is at significantly lower efficiency (83 %), with a few other lower energy runs around 91 %. The data quality in the CDL data is lower and it has larger variation of the common properties (gas gain, diffusion) as compared to the CAST dataset.

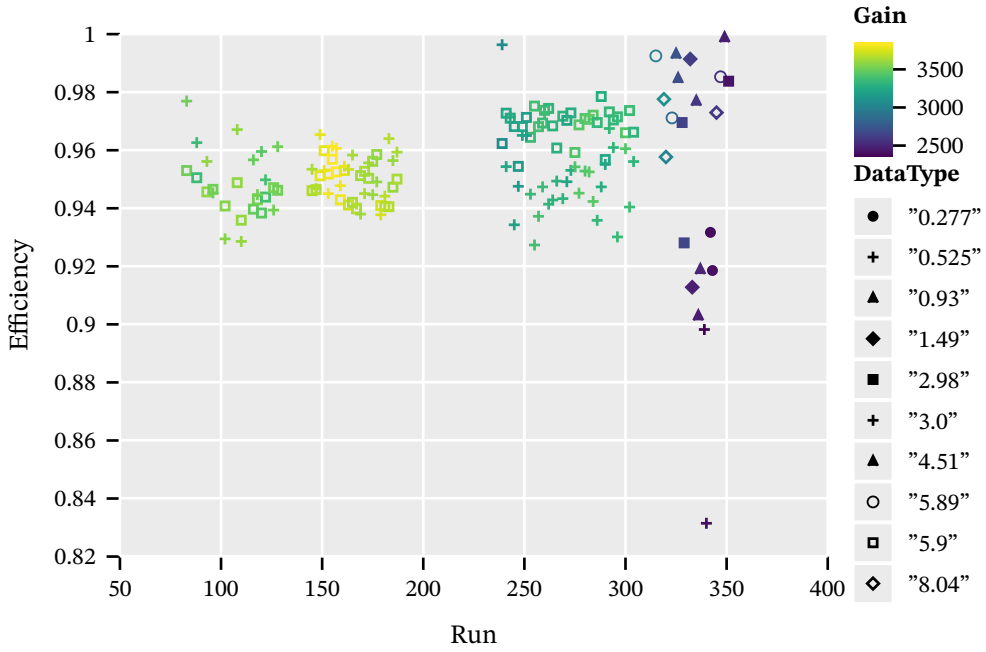


Figure 11.16.: Effective efficiencies obtained using the MLP for a target $\varepsilon = 95\,\%$ software efficiency. Runs above run 305 are CDL calibration runs. Different symbols represent different target fluorescence lines. The ^{55}Fe escape peak is indicated at 3 keV in contrast to the 2.98 keV silver line, due to different physical properties. The photopeak corresponds to `5.9` and the equivalent CDL Mn target to `5.89`.

11.4.8. Background rate using MLP

Applying the MLP cut as explained leads to background rates as presented in fig. 11.17, where the MLP is compared to the $\ln\mathcal{L}$ cut ($\varepsilon = 80\,\%$) at a similar software efficiency $\varepsilon = 84.7\,\%$ as well as at a significantly higher efficiency of $\varepsilon = 94.4\,\%$. Note the efficiencies are effective efficiencies as explained in sec. 11.4.7.

The background suppression is significantly improved in particular at lower energies, implying other cluster properties provide better separation at those than the three inputs used for the likelihood cut. The mean background rates between 0.2 to 8 keV for each of these are [27]:

$$b_{\ln\mathcal{L}@80\,\%} = 2.061(96) \times 10^{-5}\,\mathrm{keV}^{-1}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$$
$$b_{\mathrm{MLP}@85\,\%} = 1.565(84) \times 10^{-5}\,\mathrm{keV}^{-1}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$$
$$b_{\mathrm{MLP}@95\,\%} = 2.016(95) \times 10^{-5}\,\mathrm{keV}^{-1}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$$

The most important aspect is that the MLP allows to achieve very comparable background rates at significantly higher efficiencies. Accepting lower signal efficiencies does not come with a noticeable improvement, implying that for the remaining clusters the distinction between background and X-ray properties is very small (and some remaining clusters surely are of X-ray origin).

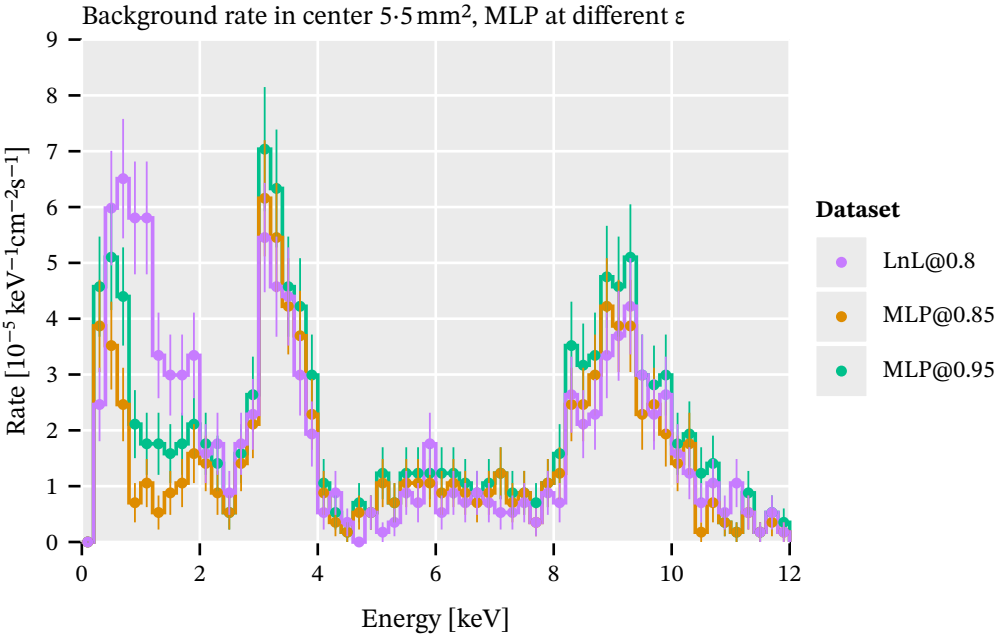


Figure 11.17.: Comparison of the background rate in the center region of the MLP at different efficiencies and the standard $\ln\mathcal{L}$. The MLP improves most at low energies and achieves comparable or better rates at higher efficiencies.

[27] The number quoted here for $\ln\mathcal{L}$ at 80 % differs slightly from sec. 11.3, because the lower energy range is 0.2 keV here.

11.5. Additional detector features as vetoes

Next we will cover how the additional detector features can be used as vetoes to suppress even more background. We start by looking at the scintillators in sec. 11.5.1. Then we consider the FADC as a veto based on time evolution of the events in sec. 11.5.2. Finally, we consider the outer GridPix ring as further geometric vetoes in sec. 11.5.3 and sec. 11.5.4 in the form of the 'septem veto' and the 'line veto'.

Note, in the context of these detector vetoes we will generally apply these vetoes on top of the the $\varepsilon = 80\,\%\ \ln\mathcal{L}$ method. We will later come back to the MLP in sec. 11.6.

11.5.1. Scintillators as vetoes

As introduced in theory section 5.1.4 one problematic source of background is X-ray fluorescence created due to abundant muons interacting with material of – or close to – the detector. The resulting X-rays, if detected, are indistinguishable from those X-rays originating from an axion (or other particle of study). And given the required background levels, cosmic induced X-ray fluorescence plays a significant role in the datasets. Scintillators (ideally 4π around the whole detector setup) can be very helpful to reduce the influence of such background events by 'tagging' certain events. For a fully encapsulated detector, any (at least) muon induced X-ray would be preceded by a signal in one (or multiple) scintillators. As such, if the time t_s between the scintillator trigger and the time of activity recorded with the GridPix is small enough, the two are likely to be in real coincidence.

In the setup used at CAST with a 42 cm times 82 cm scintillator paddle (see sec. 6.7), about $\sim 30\,\text{cm}$ above the detector center – and a $\cos^2(\theta)$ distribution for muons – a significant fraction of muons should be tagged. Similarly, the second scintillator on the Septemboard detector, the small SiPM behind the readout area should trigger precisely in those cases where a muon traverses the detector from the front or back such that the muon track is orthogonal to the readout plane.

The term 'non trivial trigger' in the following indicates events in which a scintillator triggered and the number of clock cycles to the GridPix readout was larger than 0 (scintillator had a trigger in the first place) and smaller than 300. The latter cut is somewhat arbitrary, as the idea is two things: 1. no clock cycles of 4095 (indicates clock ran over) and 2. the physics involved leading to coincidences typically takes place on time scales shorter than 100 clock cycles $= 2.5\,\mu\text{s}$ (see sec. 6.7). Anything above should just be a random coincidence. 300 is therefore just chosen to have a buffer to where physical clock cycles start.

During the Run-3 data taking period a total of 69 243 non trivial veto paddle triggers were recorded. [28] The distribution of the clock cycles after which the GridPix readout happened is shown in fig.

[28] A ball park estimate yields a coverage of about 35 ° around the zenith. Assuming $\cos^2(\theta)$ muons it covers about 68 % of muons in that axis. With a rate of $\sim 1\,\text{cm}^{-2}\,\text{min}^{-1}$, 1125 h of data with scintillator and 4.2 cm^2 of active area in front

11.18 on the right. The narrow peak at 255 clock cycles seems to be some kind of artifact, potentially a firmware bug. The corresponding GridPix data was investigated and there is nothing unusual about it (neither cluster properties nor individual events). The source therefore remains unclear, but a physical origin is extremely unlikely as the peak is exactly one clock cycle wide (unrealistic for a physical process in this context) and coincidentally exactly 255 (`0xFF` in hexadecimal), hinting towards a counting issue in the firmware. The actual distribution looks as expected, being a more or less flat distribution corresponding to muons traversing at different distances to the readout. The real distribution does not start at exactly 0 clock cycles due to inherent processing delays and even close tracks requiring some time to drift to the readout and activating the FADC. Further, geometric effects play a role. Very close to the grid, only perfectly parallel tracks can achieve low clock cycles, but the further away different angles contribute to the same times.

The SiPM recorded 4298 non trivial triggers in the same time, which are shown in fig. 11.18 on the left. [29] Also this distribution looks more or less as expected, showing a peak towards low clock cycles (typical ionization and therefore similar times to accumulate enough charge to trigger) with a tail for less and less ionizing tracks. The same physical cutoff as in the veto paddle distribution is visible corresponding to the full 3 cm of drift time.

Both of these distributions and the physically motivated cutoffs in clock cycles motivate a scintillator veto at clock cycles somewhere above 60 to 70 clock cycles. To be on the conservative side and because random coincidences are very unlikely (on the time scales of several clock cycles; picking a value slightly larger implies a negligible dead time), a scintillator veto cut of 150 clock cycles was chosen.

The resulting improvement of the background rate is shown in fig. 11.19, albeit only for the end of 2018 (Run-3) data as the scintillator trigger was not working correctly in Run-2 (as mentioned in sec. 9.5). The biggest improvements can be seen in the 3 keV and 8 keV peaks, both of which are likely X-ray fluorescence ($\mathrm{Ar}_{K\alpha}$ 3 keV & $\mathrm{Cu}_{K\alpha}$ 8 keV) and orthogonal muons (> 8 keV). Arguably the improvements could be bigger, but the efficiency of the scintillator was not ideal, resulting in some likely muon induced X-rays to remain untagged. Lastly, the coverage of the scintillators is leaving large angular areas without a possibility to be tagged.

For a future detector an almost 4π scintillator setup and correctly calibrated and tested scintillators are an extremely valuable upgrade.

of center GridPix roughly 194 000 muons are expected. About 70 000 non trivial triggers were recorded. Efficiency of $\sim 35\,\%$.

[29] Estimating the number of expected muons for the SiPM is much more difficult, because there is little literature on the actual muon rate at angles close to 90 °. The extended version of this thesis contains some calculations, which try to estimate energy distribution and rate under these angles in the detector based on numerically transporting muons from the upper atmosphere to the surface undergoing relativistic effects, changing atmospheric pressures and the related energy loss per distance.

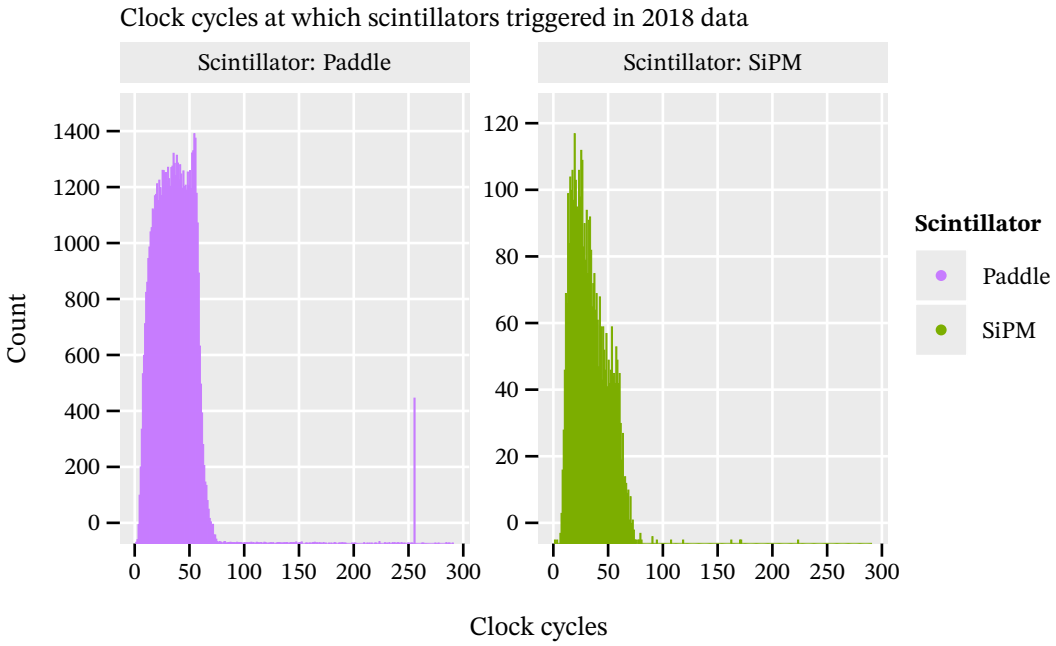


Figure 11.18.: Clock cycle distributions of both scintillators of the end of 2018 data. The data is filtered to all non-trivial triggers (non zero and less than 300; there are individual random coincidences in values up to 4095 where all triggers whose counter overran are). The origin of the peak at 255 clocks in the data of the paddle is unclear.

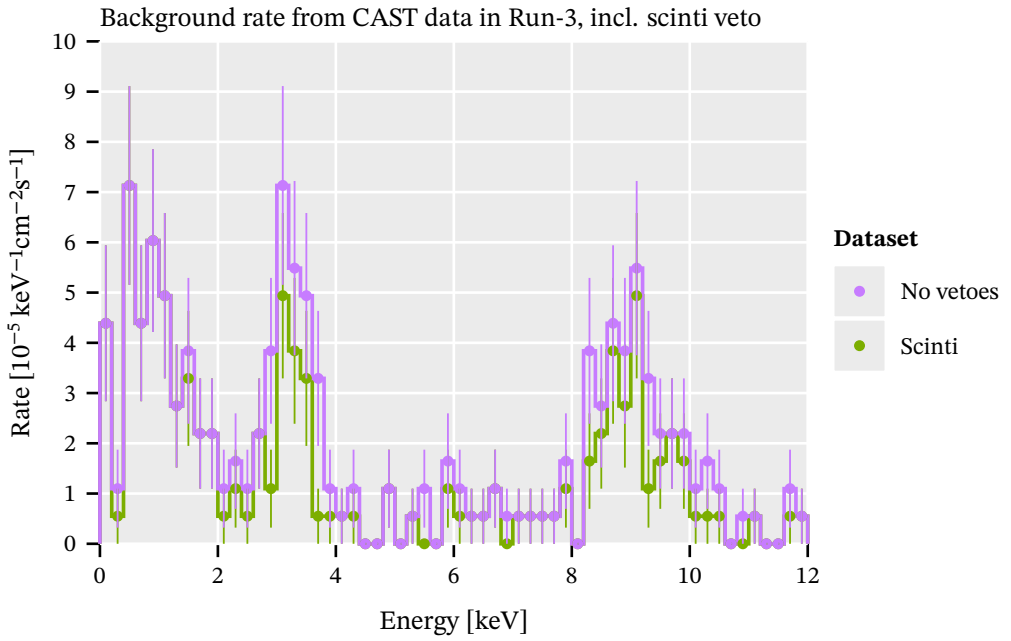


Figure 11.19.: Background rate based on the Run-3 CAST data achieved by the addition of a scintillator cut veto of 3.75 µs ($150 clock cycles) for any cluster that initially passes the log likelihood cut. The biggest improvements can be seen in the 3 keV and 8 keV peaks, both of which are likely X-ray fluorescence (Cu & Ar; both energies) and orthogonal muons (> 8 keV).

11.5.2. FADC veto

As previously mentioned in 6.8 the FADC not only serves as a trigger for the readout and reference time for the scintillator triggers. Because of its high temporal resolution it can in principle act as a veto of its own by providing insight into the longitudinal cluster shape.

A cluster drifting towards the readout and finally through the grid induces a voltage measured by the FADC. As such the length of the FADC signal is a function of the time it takes the cluster to drift 'through' the grid. The kind of orthogonal muon events that should be triggered by the SiPM as explained in the previous section 11.5.1 for example should also be detectable by the FADC in the form of longer signal rise times than typical in an X-ray.

From gaseous detector physics theory we can estimate the typical sizes and therefore expected signal rise times for an X-ray if we know the gas of our detector. For the 1050 mbar, 97.7/2.3 % $\mathrm{Ar/iC_4H_{10}}$ mixture used with a 500 V cm^{-1} drift field in the Septemboard detector at CAST, the relevant parameters are [30]

- drift velocity $v = 2.28\,\mathrm{cm\,\mu s^{-1}}$
- transverse diffusion $\sigma_T = 670\,\mu\mathrm{m\,cm^{-1/2}}$
- longitudinal diffusion $\sigma_L = 270\,\mu\mathrm{m\,cm^{-1/2}}$

As the detector has a height of 3 cm we expect a typical X-ray interacting close to the cathode to form a cluster of $\sqrt{3\,\mathrm{cm}} \cdot 670\,\mu\mathrm{m\,cm^{-1/2}} \approx 1160.5\,\mu\mathrm{m}$ transverse extent and $\sqrt{3\,\mathrm{cm}} \cdot 270\,\mu\mathrm{m\,cm^{-1/2}} \approx 467.5\,\mu\mathrm{m}$ in longitudinal size, where this corresponds to a 1σ environment. To get a measure for the cluster size, a rough estimate for an upper limit is a 3σ distance away from the center in each direction. For the transverse size it leads to about 7 mm and in longitudinal about 2.8 mm. From the CAST ^{55}Fe data we see a peak at around 6 mm of transverse cluster size along the longer axis, which matches well with our expectation (see appendix K.3 for the length data). From the drift velocity and the upper bound on the longitudinal cluster size we can therefore also compute an equivalent effective drift time seen by the FADC. This comes out to be

$$t = 2.8\,\mathrm{mm}/22.8\,\mathrm{mm\,\mu s^{-1}} = 0.123\,\mu\mathrm{s}$$

or about 123 ns, equivalent to 123 clock cycle of the FADC clock. Alternatively, we can compute the most likely rise time based on the known peak of the cluster length, 6 mm and the ratio of the transverse and longitudinal diffusion, $\sigma_T/\sigma_L \approx 2.5$ to end up at a time of $\frac{6\,\mathrm{mm}}{2.5} \cdot 22.8\,\mathrm{mm\,\mu s^{-1}} = 105\,\mathrm{ns}$.

[30] The properties were calculated with `Magboltz` using `Nimboltz`, a Nim interfacer for `Magboltz`. The code can be found here: `https://github.com/Vindaar/TimepixAnalysis/Tools/septemboardCastGasNimboltz` Further the numbers used here are for a temperature of 26 °C, slightly above room temperature due to the heating of the Septemboard itself into the gas.

Fig. 11.20 shows the measured rise time for the ^{55}Fe calibration data compared to the background data. The peak of the calibration rise time is at about 55 ns. While this is almost a factor of 2 smaller than the theoretical values mentioned before, this is expected as those first of all are an upper bound and secondly the ”rise time” here is not the full time due to our definition starting from 10 % below the baseline and stopping 2.5 % before the peak, shortening our time (ref. sec. 8.5.3). At the same time we see that the background data has a much longer tail towards higher clock cycle counts, as one expects. This implies that we can perform a cut on the rise time and utilize it as an additional veto. The ^{55}Fe data allows us to define a cut based on a known and desired signal efficiency. This is done by identifying a desired percentile on both ends of the distribution of a cleaned X-ray dataset. It is important to note that the cut values need to be determined for each of the used FADC amplifier settings separately. This is done automatically based on the known runs corresponding to each setting.

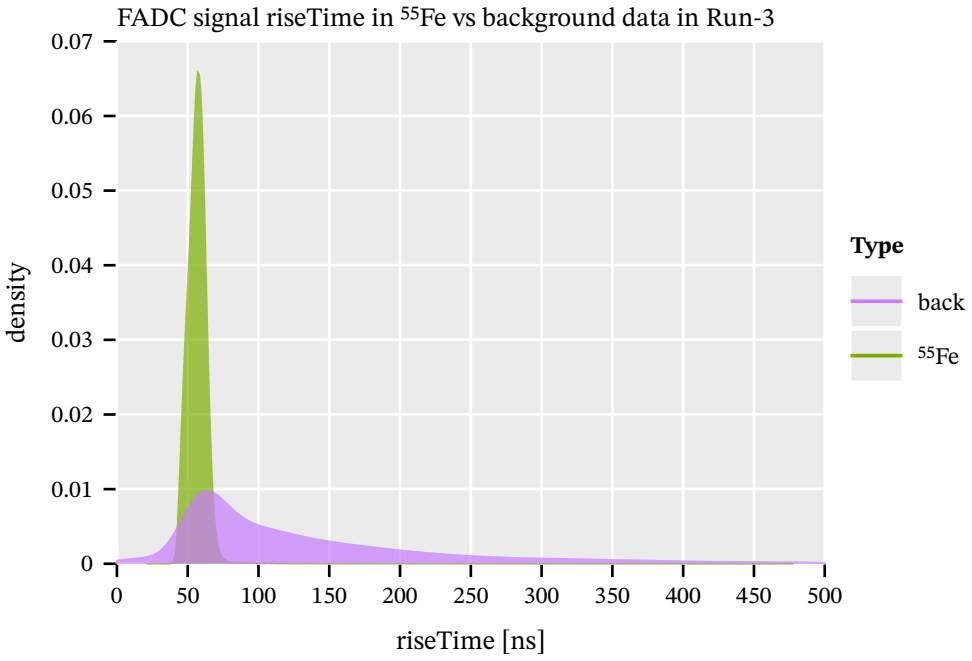


Figure 11.20.: KDE of the rise time of the FADC signals in the ^{55}Fe and background data of the CAST Run-3 dataset. The X-ray data is a single peak with a mean of about 55 ns while the background distribution is extremely wide, motivating a veto based on this data.

For the signal decay time we do a simpler cut, with less of a physical motivation. As the decay time is highly dependent on the resistance and capacitance properties of the grid, high voltage and FADC amplifier settings, we simply introduce a conservative cut in such a range as to avoid cutting away any X-rays. Any background removed is just a bonus.

Finally, the FADC is only used as a veto if the individual spectrum is not considered noisy (see sec. 10.4.1). This is determined by 4 local dominant peaks based on a peak finding algorithm and

in addition a general skewnees of the total signal larger than −0.4. The skewness is an empirical cutoff as good FADC signals typically lie at larger negative skewness values (due to them having a signal towards negative values). See appendix K.2 for a scatter plot of FADC rise times and skewness values. A very low percentage of false positives (good signals appearing 'noisy') is accepted for the certainty of not falsely rejecting noisy FADC events (as they would represent a random coincidence to the event on the center chip).

Applying the FADC veto with a target percentile of 1 from both the lower and upper end (1^{st} and 99^{th} percentiles; thus a 98 % signal efficiency) in addition to the scintillator veto presented in the previous section, results in a background rate as seen in fig. 11.21. Different percentiles (and associated efficiencies) were be tested for their impact on the expected axion limit. Lower percentiles than 99 did not yield a significant improvement in the background suppression over the resulting signal efficiency loss. This is visible due to the sharp edge of the rise time for X-rays (green) in fig. 11.20 when comparing to its overlap with background (purple).

The achieved background rate of the methods leads to a background rate of $9.03(73) \times 10^{-6}\,\text{keV}^{-1}\,\text{cm}^{-2}\,\text{s}^{-1}$ in the energy range 2 to 8 keV. And between 4 to 8 keV even $4.57(63) \times 10^{-6}\,\text{keV}^{-1}\,\text{cm}^{-2}\,\text{s}^{-1}$. The veto brings improvements across the whole energy range, in which the FADC triggers. Interestingly, in some cases even below that range. The events removed in the first two bins are events with a big spark on the upper right GridPix, which induced an X-ray like low energy cluster on the central chip and triggered the FADC. In the range at around 8 keV, likely candidates for removed events are cases of orthogonal muons that did not trigger the SiPM (but have a rise time incompatible with X-rays).

Generally, appendix K.2 contains more figures of the FADC data. Rise times, fall times, how the different FADC settings affect these and an efficiency curve of a one-sided cut on the rise time for the signal efficiency and background suppression.

11.5.3. Outer GridPix as veto - 'septem veto'

The final hardware feature that is used to improve the background rate is the outer ring of GridPixes. The size of large clusters is a significant fraction of a single GridPix. This means that depending on the cluster center position, parts of the cluster may very well be outside of the chip. While the most important area of the chip is the center area (due to the X-ray optics focusing the axion induced X-rays), misalignment and the extended nature of the 'axion image' mean that a significant portion of the chip should be as low in background as possible. Fig. 11.10(a) as we saw in sec. 11.3 shows a significant increase in cluster counts towards the edges and corners of the center GridPix. The GridPix ring can help us reduce this.

Normally the individual chips are treated separately in the analysis chain. The 'septem veto' is the name for an additional veto step, which can be optionally applied to the center chip. With it, each

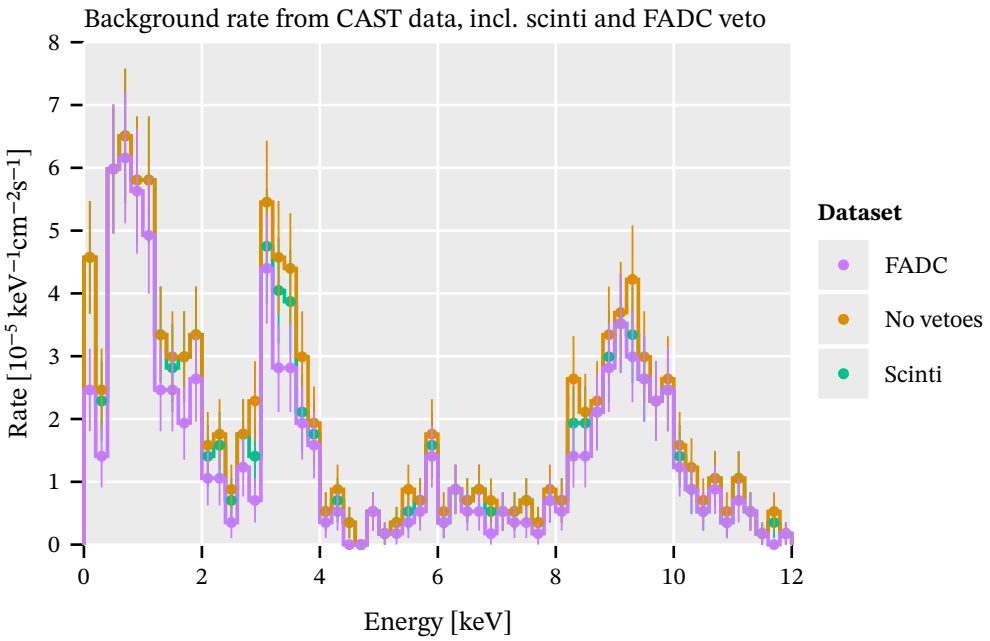


Figure 11.21.: Comparison of pure $\ln \mathcal{L}$ cuts, additional scintillator veto and finally also FADC veto at an efficiency of 98 %. In the range 2 to 8 keV a background rate of $9.03(73) \times 10^{-6}\,\mathrm{keV^{-1}\,cm^{-2}\,s^{-1}}$ is achieved and between 4 to 8 keV even $4.57(63) \times 10^{-6}\,\mathrm{keV^{-1}\,cm^{-2}\,s^{-1}}$.

cluster that is considered signal-like based on the likelihood cut (or MLP), will be analyzed in a second step. The full event is reconstructed again from the beginning, but this time as an event covering *all 7* chips. This allows the cluster finding algorithm to detect clusters beyond the center chip boundaries. After finding all clusters, the normal cluster reconstruction to compute all properties is done again. Finally, for each cluster in the event the likelihood method or MLP is applied. If now all clusters whose center is on the central chip are considered background-like, the event is vetoed, because the initial signal-like cluster turned out to be part of a larger cluster covering more than one chip.

There is a slight complication here. The layout of the septemboard includes spacing (see for example fig. 6.14 or 8.3 for the layout), which is required to mount the chips. This spacing, in addition to general lower efficiency towards the edges of a GridPix, means significant information is lost. When reconstructing the full 'septemboard events' this spacing would break the cluster finding algorithms, as the spacing might extend the distance over the cutoff criterion [31]. For this reason the cluster finding algorithm is actually performed on a 'tight layout' where the spacing has been reduced to zero. For the calculation of the geometric properties however, the clusters are transformed into the real septemboard coordinates including spacing. [32]

[31] An alternative could be to attempt to fill in the area between the chips with simulated data based on what is seen at the chip edges. But that is a complex problem. A fun (but possibly questionable) idea would be to train a diffusion model (generative AI) to predict missing data between chips.

[32] In principle this has a small impact on the calculation of the RMS of the cluster, if there is now a large gap between

In contrast to the regular single chip analysis performed before, which uses a simple, bespoke clustering algorithm (see sec. 8.4.1, 'Default'), the full 'septemboard reconstruction' uses the DBSCAN clustering algorithm. The equivalent search radius is extended a bit over the normal default (65 pixels compared to 50, unless changed in configuration). DBSCAN produces better results in terms of likely related clusters (over chip boundaries). [33]

An example that shows two clusters on the center chip, one of which was initially interpreted as a signal-like event before being vetoed by the 'septem veto', is shown in fig. 11.22. The colors indicate the clusters each pixel belongs to according to the cluster finding algorithm. As the chips are treated separately initially, there are two clusters found on the center chip. The green and purple cluster "portions" of the center chip. The cyan part passes the likelihood cut initially, which triggers the 'septem veto' (X-rays at such low energies are much less spherical on average; same diffusion, but less electrons). A full 7 GridPix event reconstruction shows the additional parts of the two clusters. The cyan cluster is finally rejected. It is a good example, as it shows a cluster that is still relatively close to the center, and yet still 'connects' to another chip.

The background rate with the septem veto is shown in fig. 11.23, where we see that most of the improvement is in the lower energy range $< 2\,\text{keV}$. This is the *most important region for the solar axion flux for the axion-electron coupling*. Looking at the mean background rate in intervals of interest, between 2 to 8 keV it is $8.03(69) \times 10^{-6}\,\text{keV}^{-1}\,\text{cm}^{-2}\,\text{s}^{-1}$ and 4 to 8 keV $= 4.05(60) \times 10^{-6}\,\text{keV}^{-1}\,\text{cm}^{-2}\,\text{s}^{-1}$. And even in the full range down to 0 keV the mean is in the 10^{-6} range, $9.54(65) \times 10^{-6}\,\text{keV}^{-1}\,\text{cm}^{-2}\,\text{s}^{-1}$.

11.5.4. 'Line veto'

There is one more further optional veto, dubbed the 'line veto'. It checks whether there are clusters on the outer chips whose long axis "points at" clusters on the center chip (passing the likelihood or MLP cut). The idea being that there is a high chance that such clusters are correlated, especially because ionization is an inherently statistical process. It can be used in addition to the 'septem veto' or standalone. The approach uses the same general approach as the septem veto by reconstructing the full events as 'septemboard events' again. The difference to the septem veto is that it is not reliant on the cluster finding search radius.

The center cluster that initially passed the likelihood or MLP cut will be rejected if the long axis of the outer chip cluster cuts within $3 \cdot \left(\frac{\text{RMS}_T + \text{RMS}_L}{2}\right)$. In other words within 3σ of the mean (along long

two parts of the 'same' cluster. However, as we don't really rely on the properties too much it is of no real concern. Technically we still calculate the $\ln \mathcal{L}$ of each cluster, but any cluster passing the $\ln \mathcal{L}$ cut before being added to a larger cluster almost certainly won't pass it afterwards. The slight bias in the RMS won't change that.

[33] The downside is that DBSCAN is significantly slower, which is part of the reason it is not utilized for the initial reconstruction.

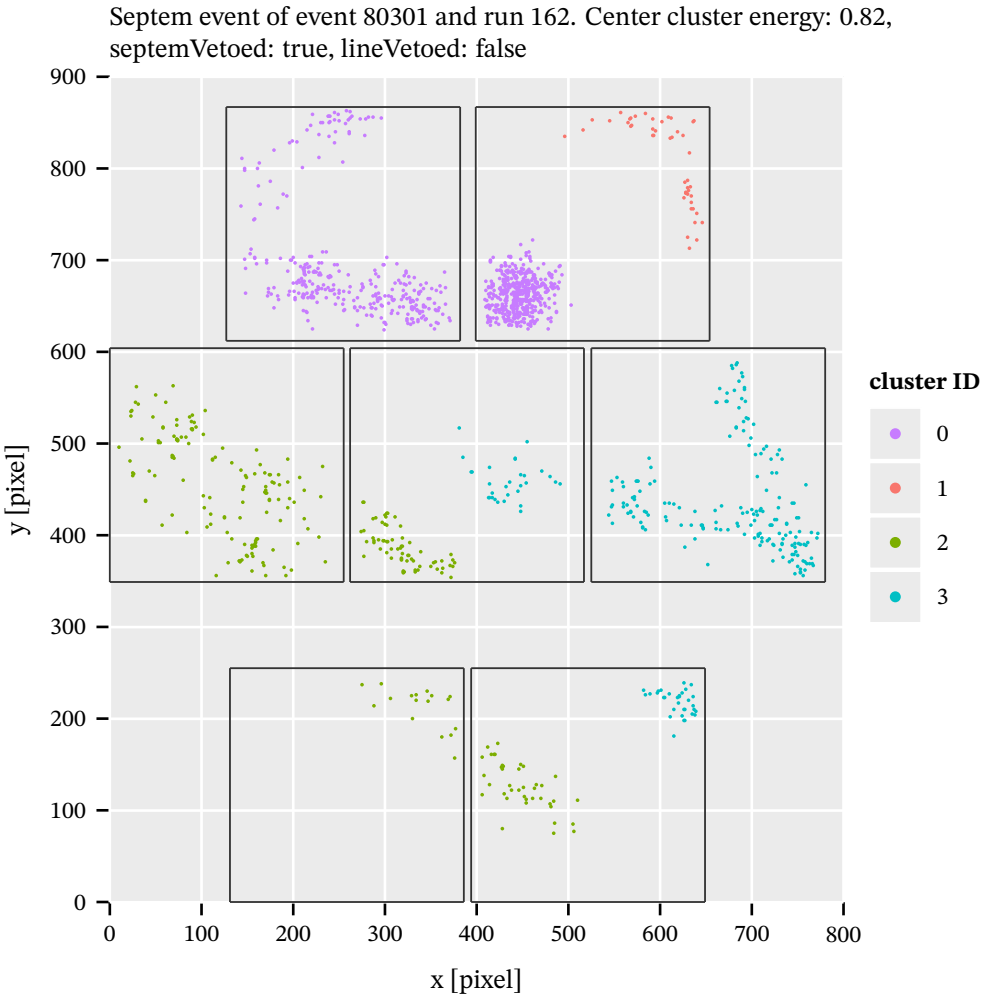


Figure 11.22.: An example event showing all 7 GridPix of the CAST GridPix1 detector. The outlines are the boundaries of each chip and the color of each point indicates the cluster which it is part of according to the cluster finder. Initially the cyan cluster (center chip portion) passes the log likelihood cut (i.e. is signal like), but is vetoed by the 'septem veto' as there are more pixels outside the center chip that are part of this cluster. The green cluster in the bottom left of the center chip is in addition a good example of how in particular cutting a track in the corners leads to a much more spherical cluster.

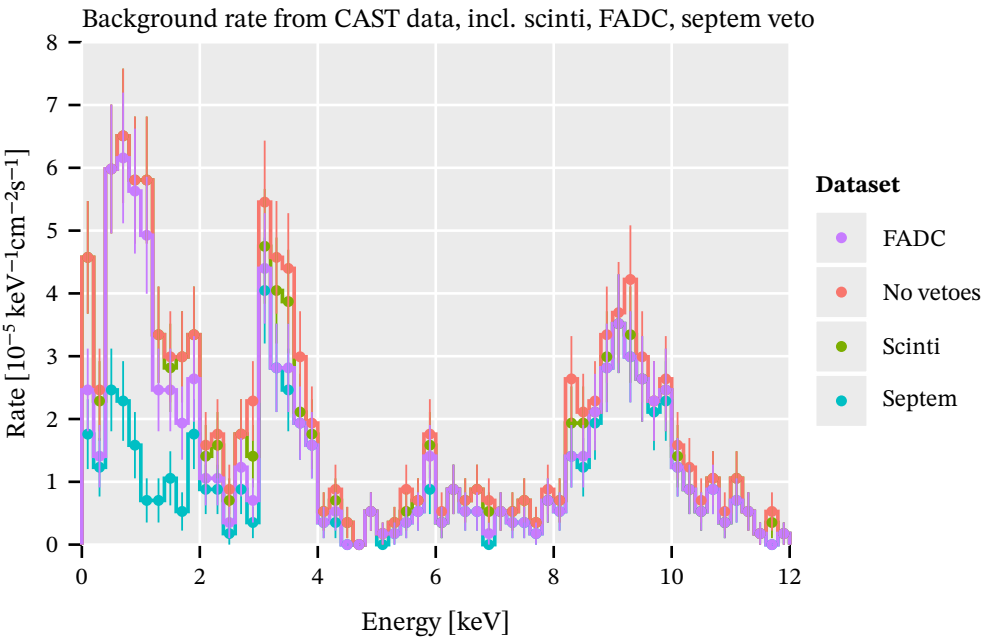


Figure 11.23.: Background rate achieved based on the 'septem veto' (in addition to the scintillator cut and FADC veto) for the full 2017/18 dataset within the center 5 × 5 mm². Significant improvement in the < 2 keV range, which is most important for axion-electron coupling solar flux. The background rate between 2 to 8 keV is $8.03(69) \times 10^{-6}$ keV^{-1} cm^{-2} s^{-1} and 4 to 8 keV = $4.05(60) \times 10^{-6}$ keV^{-1} cm^{-2} s^{-1}.

and short axis) standard distribution of the pixels [34].

If desired not every outer chip cluster is considered for the 'line veto'. An ε_{cut} parameter can be adjusted to only allow those clusters to contribute with an eccentricity larger than ε_{cut}. The value of this cutoff impacts the efficiency, but also the expected random coincidence rate of the veto. More on that in sec. 11.5.5.

An example of an event being vetoed by the 'line veto' is shown in fig. 11.24. The black circle around the center chip cluster indicates the radius in which the orange line (extension of the long axis of the top, green cluster) needs to cut the center cluster.

The achieved background rate, see fig. 11.25, goes well into the 10^{-6} range with this veto over the whole energy range, as the influence is largest at low energies. The rate comes out to $7.92(59) \times 10^{-6}$ keV^{-1} cm^{-2} s^{-} over the whole range up to 8 keV, to $7.56(67) \times 10^{-6}$ keV^{-1} cm^{-2} s^{-1} if starting from 2 keV and $3.78(58) \times 10^{-6}$ keV^{-} if starting from 4 keV.

11.5.5. Estimating the random coincidence rate of the septem & line veto

One potential issue with the septem and line veto is that the shutter times we ran with at CAST are very long (> 2 s), but only the center chip is triggered by the FADC. This means that the outer chips

[34] This is roughly the radius of the cluster going by most extended active pixels of the cluster.

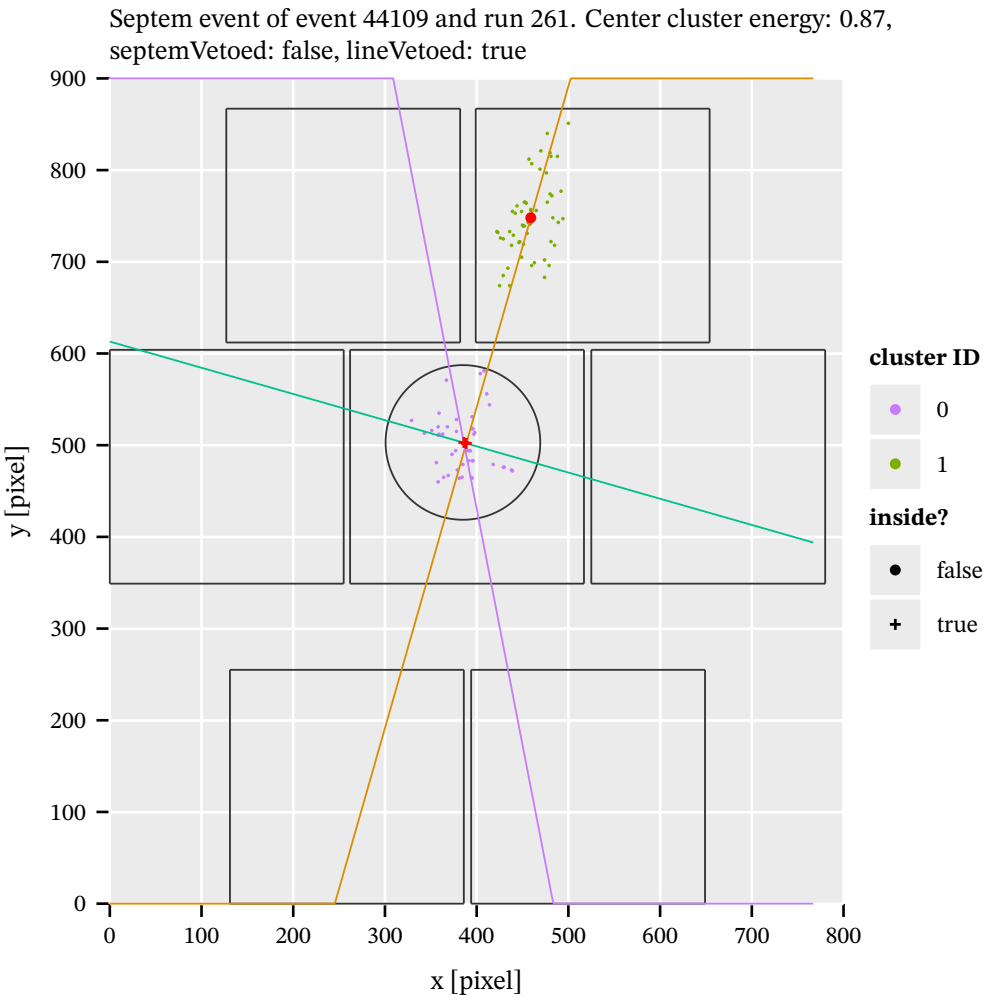


Figure 11.24.: Example event, which highlights the use case of the 'line veto'. The green cluster in the upper chip is eccentric and its long axis 'points' towards the purple center cluster (which initially passes the $\ln \mathcal{L}$ cut). The black circle is a measure for the radius of the center cluster. If the line of the eccentric cluster cuts the circle, the cluster is vetoed.

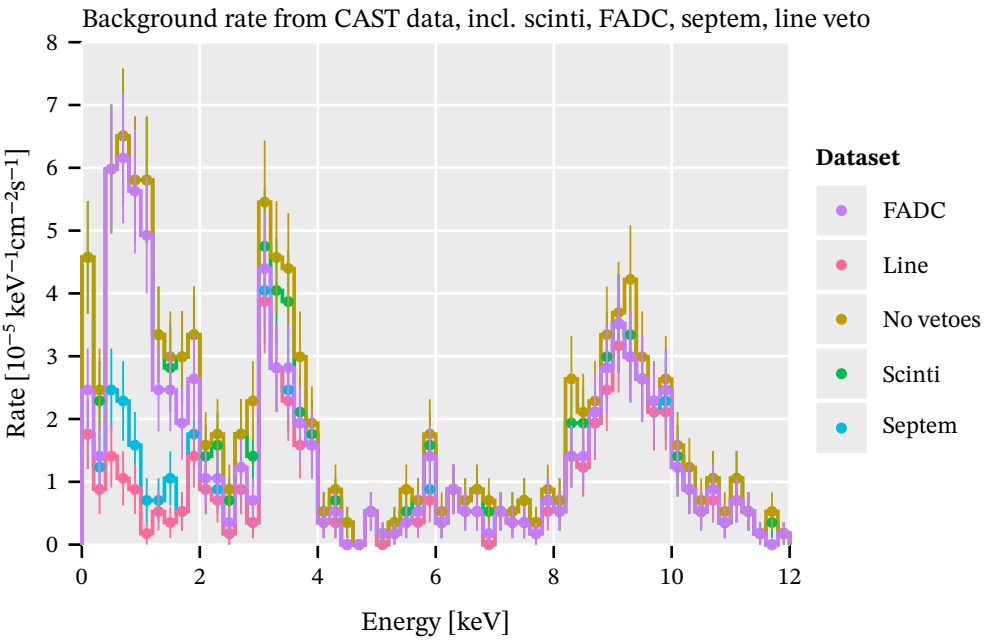


Figure 11.25.: Background rate of all CAST 2017/18 data in the center $5 \times 5\,\mathrm{mm}^2$ using all vetoes including the line veto. Improvements are seen especially at low energies. The mean background rate in the full range below 8 keV is $7.92(59) \times 10^{-6}\,\mathrm{keV}^{-1}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$.

can record cluster data not correlated to what the center chip sees. When applying one of these two vetoes the chance for random coincidence might be non negligible. This random coincidence rate needs to be treated as a dead time for the detector, reducing the effective solar tracking time available.

In order to estimate this we bootstrap fake events with guaranteed random coincidence. For each data taking run, we take events with clusters on the center chip, which pass the likelihood or MLP cuts and combine these with data from the outer chips from different events. The outer GridPix ring data is sampled from *all* events in that run, including empty events as otherwise we would bias the coincidence rate. The vetoes (either only septem veto, only line veto or both) are then applied to this bootstrapped dataset. The fraction of rejected events due to the veto is the random coincidence rate, because there is no physical correlation between center cluster and outer GridPix ring clusters.

By default 2000 such events are generated for each run. The random coincidence rate is the mean of all obtained values. Further, this can be used to study the effectiveness of the eccentricity cutoff ε_{cut} mentioned for the line veto. It is expected that a stricter eccentricity cut should yield less random coincidences. We can then compare the obtained random coincidence rates to the real fractions of events removed by the same veto setup. The ratio of real effectiveness to random coincidence rate is a measure of the potency of the veto setup.

Fig. 11.26 shows how the fraction of passing events changes for the line veto as a function of the eccentricity cluster cutoff. The cases of line veto only for fake and real data, as well as septem veto

plus line veto for the same are shown. We can see that the fraction that passes the veto setups (y axis) drops the further we go towards a low eccentricity cut (x axis). For the real data (`Real` suffix in the legend) the drop is faster than for fake boostrapped data (`Fake` suffix in the legend), which means that we can use the lowest eccentricity cut possible (effectively disabling the cut at $\varepsilon_{\text{cut}} = 1.0$). In any case, the difference is very minor independent of the cutoff value. The septem veto without line veto is also shown in the plot (`SeptemFake` and `SeptemReal`) with only a single point at $\varepsilon_{\text{cut}} = 1.0$ for completeness.

Table 11.7 shows the percentages of clusters left over after the line veto (at $\varepsilon_{\text{cut}} = 1.0$) and septem veto setups, both for real data (column 'Real') and bootstrapped fake data (column 'Fake'). For fake data this corresponds to the random coincidence rate of the veto setup (strictly speaking $1 - x$ of course). For real data it is the real percentage of clusters left, meaning the vetoes achieve reduction to $\frac{1}{10}$ of background, depending on setup. The random coincidence rates are between 78.6 % in case both outer GridPix vetoes are used, 83.1 % for the septem veto and 85.4 % for the line veto only. The line veto has lower random coincidence rate, but by itself is also less efficient.

This effect on the signal efficiency is one of the major reasons the choice of vetoes is not finalized outside the context of a limit calculation.

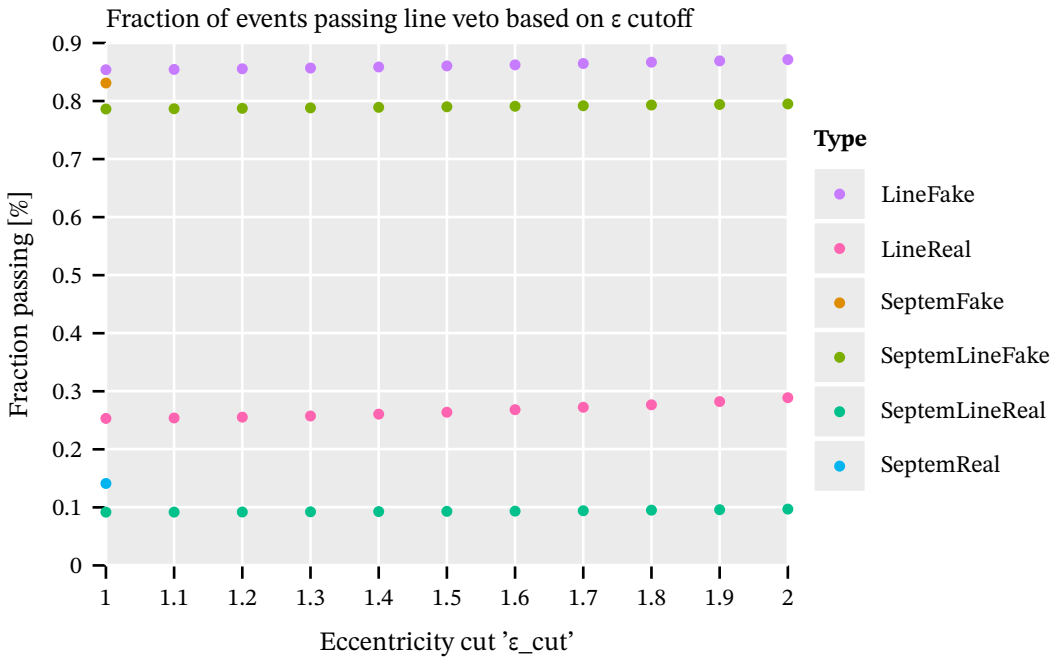


Figure 11.26.: Fraction of events in Run-3 data, which pass (i.e. not rejected) the line veto depending on the eccentricity cut used, which decides how eccentric a cluster needs to be in order to be used for the veto. 'Real' and 'Fake' suffixes refer to application of the vetoes to the bootstrapped fake data discussed or real data. For fake data the percentage corresponds to the random coincidence rate. Septem veto without line veto shown as single points (`SeptemFake/Real`).

Septem veto	Line veto	Real [%]	Fake [%]
y	n	14.12	83.11
n	y	25.32	85.39
y	y	9.17	78.63

Table 11.7.: Overview of the percentages of clusters left after the line veto and and septem veto setup combinations (y = activated, n = not activated) for bootstrapped ('Fake') data and real background data. The percentages for fake data correspond to the random coincidence rate.

11.6. Background rates of combined vetoes and efficiencies

Having discussed the two main classifiers (likelihood and MLP) as well as all the different possible vetoes, let us now recap. We will look at background rates obtained for different veto setups and corresponding signal efficiencies. In addition, we will compare with fig. 11.10 – the background clusters left over the entire chip and local background suppression – when using all vetoes.

For a comparison of the background rates achievable in the center $5 \times 5\,\mathrm{mm}^2$ region using the $\ln\mathcal{L}$ method and the MLP, see fig. 11.27. As already expected from the MLP performance in sec. 11.4.8, it achieves comparable performance at significantly higher signal efficiency. This will be a key feature in the limit calculation to increase the total signal efficiency. The center region does not tell the whole story though, as the background is not homogeneous.

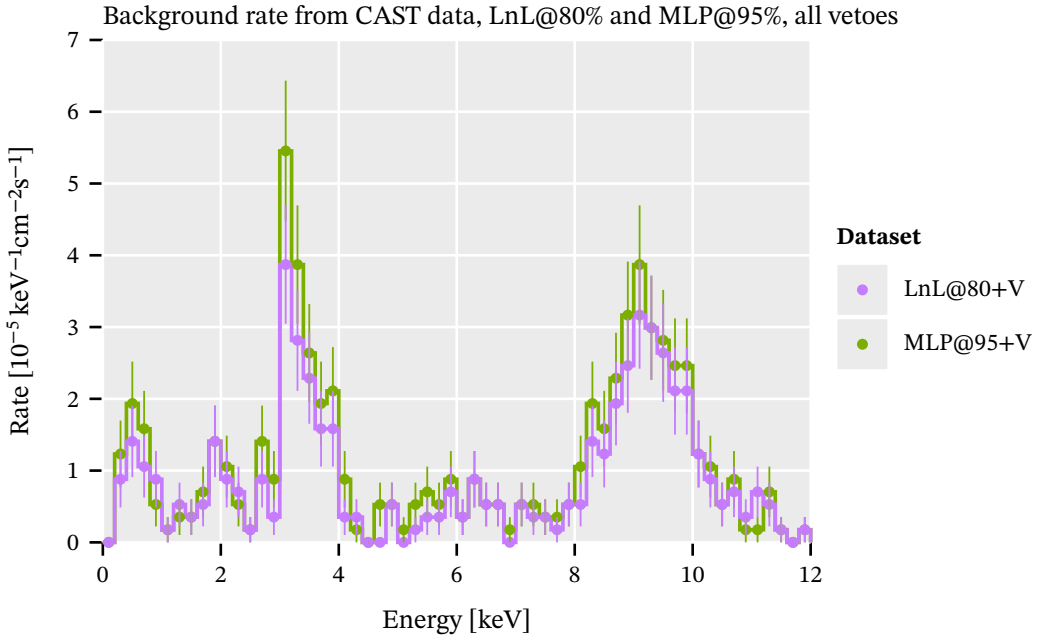
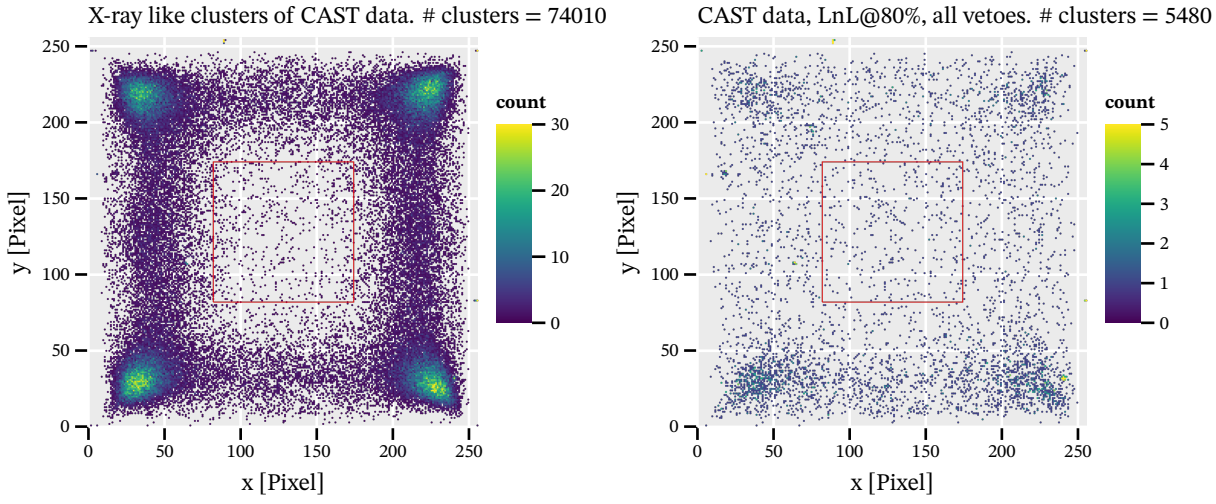


Figure 11.27.: Comparison of the $\ln\mathcal{L}$ at $\varepsilon_{\text{eff}} = 80\,\%$ and the MLP at $\varepsilon_{\text{eff}} = 95\,\%$ signal efficiency with all added vetoes. For the total combined efficiencies see tab. 11.8.

Utilizing all vetoes in addition to the $\ln \mathcal{L}$ or MLP classifier alone significantly improves the background rejection over the entire chip. See fig. 11.28 comparing all remaining X-ray like background cluster centers, without any vetoes and with all vetoes, using the $\ln \mathcal{L}$ method at $\varepsilon_{\text{eff}} = 80\,\%$. While the center background improves, the largest improvements are seen towards the edges and the corners. From a highly non uniform background rate in fig. 11.28(a), the vetoes produce an almost homogeneous background in 11.28(b). In total they yield a ~ 13.5 fold reduction in background. Fig. 11.29 visualizes the improvements by showing the local background suppression, as a factor of the clusters left in each tile over the total number of clusters in the raw data. At closer inspection it is visible that the background improvements are slightly less near the bottom edge of the chip. This is because the spacing to the bottom row of the Septemboard is larger, decreasing the likelihood of detecting a larger cluster.

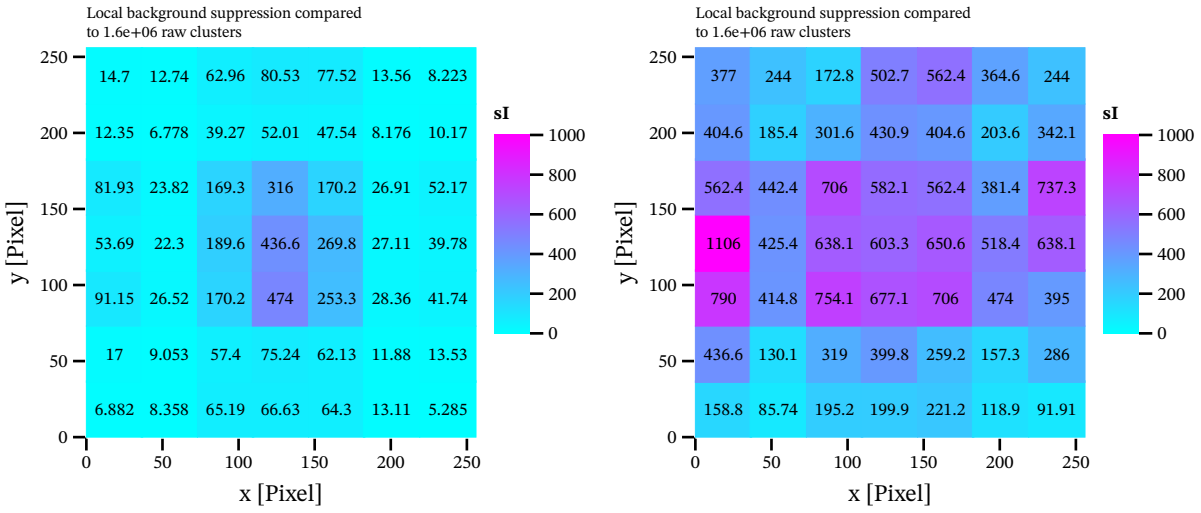


(a) No vetoes

(b) All vetoes

Figure 11.28.: Cluster centers of all X-ray like clusters in the 2017/18 CAST background data. (a) shows the remaining data for the $\ln \mathcal{L}$ method without any vetoes, (b) includes all vetoes. The vetoes lead to a dramatic reduction in background especially outside the center $5 \times 5\,\text{mm}^2$.

Tab. 11.8 shows a comparison of many different possible combinations of classifier, signal efficiency ε_{eff}, choice of vetoes and total efficiency $\varepsilon_{\text{total}}$. The 'Rate' column is the mean background rate in the range 0.2 to 8 keV in the center $5 \times 5\,\text{mm}^2$ region. The $\ln \mathcal{L}$ produces the lowest background rates, at the cost of low total efficiencies. Generally, many different setups reach into below the $10^{-5}\,\text{keV}^{-1}\,\text{cm}^{-2}\,\text{s}^{-1}$ level in the center region, while even large sacrifices in signal efficiency do not yield significantly lower rates. This implies the detector is fundamentally limited at this point in its design. The likely causes being radioactive impurities in the detector material, imperfect cosmic vetoes and lack of better time information in the events. See appendix L, in particular tab. L.1 for the same table with background rates over the entire chip.



(a) No vetoes (b) All vetoes

Figure 11.29.: Local background suppression of the $\ln \mathcal{L}$ method compared to the raw 2017/18 CAST data. (a) is the suppression without vetoes, (b) includes all vetoes.

The choice for the optimal setup purely based on background rate and total efficiency is difficult, especially because reducing the background to a single number is flawed [35]. For this reason we will compute expected limits for each of these setups. The setup yielding the best expected limit will then be used for the calculation of solar tracking candidates and observed limit.

[35] Flawed, because a) the background is not homogeneous over the entire chip and the effect of vetoes outside the center region is not visible and b) because the most important range for axion searches is between 0.5 to 5 keV (depending on the model).

Classifier	ε_{eff}	Scinti	FADC	Septem	Line	$\varepsilon_{\text{total}}$	Rate [keV^{-1} cm^{-2} s^{-1}]
LnL	0.700	true	true	true	true	0.503	$6.90(56) \times 10^{-6}$
LnL	0.700	true	true	false	true	0.590	$7.31(57) \times 10^{-6}$
LnL	0.800	true	true	true	true	0.574	$7.67(59) \times 10^{-6}$
LnL	0.800	true	true	false	true	0.674	$8.07(60) \times 10^{-6}$
MLP	0.865	true	true	true	true	0.621	$8.21(61) \times 10^{-6}$
MLP	0.865	true	true	false	true	0.729	$8.84(63) \times 10^{-6}$
MLP	0.912	true	true	true	true	0.655	$9.11(64) \times 10^{-6}$
LnL	0.800	true	true	true	false	0.615	$9.34(65) \times 10^{-6}$
LnL	0.900	true	true	true	true	0.646	$9.38(65) \times 10^{-6}$
MLP	0.912	true	true	false	true	0.769	$9.70(66) \times 10^{-6}$
LnL	0.900	true	true	false	true	0.759	$9.97(67) \times 10^{-6}$
MLP	0.957	true	true	true	true	0.687	$1.001(67) \times 10^{-5}$
MLP	0.957	true	true	false	true	0.807	$1.056(69) \times 10^{-5}$
MLP	0.983	true	true	true	true	0.706	$1.101(70) \times 10^{-5}$
MLP	0.983	true	true	false	true	0.829	$1.164(72) \times 10^{-5}$
LnL	0.800	true	true	false	false	0.784	$1.556(84) \times 10^{-5}$
MLP	0.865	false	false	false	false	0.865	$1.565(84) \times 10^{-5}$
LnL	0.700	false	false	false	false	0.700	$1.655(86) \times 10^{-5}$
MLP	0.912	false	false	false	false	0.912	$1.746(89) \times 10^{-5}$
LnL	0.800	true	false	false	false	0.800	$1.913(93) \times 10^{-5}$
MLP	0.957	false	false	false	false	0.957	$2.016(95) \times 10^{-5}$
LnL	0.800	false	false	false	false	0.800	$2.061(96) \times 10^{-5}$
MLP	0.983	false	false	false	false	0.983	$2.39(10) \times 10^{-5}$
LnL	0.900	false	false	false	false	0.900	$2.76(11) \times 10^{-5}$

Table 11.8.: Overview of different classifier, veto and software efficiency ε_{eff} setups yielding different total efficiencies $\varepsilon_{\text{total}}$. The background rate shown in the 'Rate' column is between 0.2 to 8 keV for the center $5 \times 5\,\text{mm}^2$ region.

12. Limit calculation Limit

Contents

In this chapter we will introduce a generic limit calculation method, which can be used to compute limits on different axion or ALP coupling constants.

The first part of this chapter focuses on the more theoretical and conceptual aspects of our limit calculation methods. The second half discusses our inputs in detail and shows our expected and observed limits.

We will start with an introduction of the method itself, a Bayesian extended likelihood approach, sec. 12.1. Step by step we will introduce the likelihood function we use (sec. 12.2), what the individual pieces are and how likelihood values are computed (sec. 12.3) and how a limit is computed from such a likelihood function, sec. 12.4. Then we introduce our approach to compute an expected limit by sampling toy candidates [1], sec. 12.5. After this we will extend our approach in sec. 12.6 to include systematic uncertainties. Due to added complexity in evaluating the thus produced likelihood function, we discuss our Markov Chain Monte Carlo (MCMC) approach to evaluate the likelihood function in sec. 12.7. This concludes the first half of the chapter. Please look into Lista's book [140] if you would like more details about Bayesian limit calculations involving nuisance parameters. Barlow [36] and Cowan [66] are also recommended for general statistics concepts.

From here we introduce all ingredients entering the likelihood function in detail, sec. 12.8. Next we discuss our systematics, sec. 12.9, after which we explain our MCMC approach in more detail (number of chains, parameters bounds, starting parameters etc.), sec. 12.10.

At this point we can finally tie everything together and discuss the expected limits obtained for a variety of different classifier and veto choices, sec. 12.11. For the best performing setup – the one yielding the best expected limit – we present our axion candidates, 12.12. Based on these we present our observed limit in sec. 12.13. Finally, in sec. 12.14 we briefly consider two other coupling constants, the axion-photon coupling $g_{a\gamma}$ and the chameleon coupling β_γ.

12.1. Limit method - introduction

We start with a few words on the terminology we use and what we have in mind when we talk about 'limits'.

Context and terminology An experiment tries to detect a new phenomenon of the kind where you expect very little signal compared to background sources. We have a dataset in which the experiment is *sensitive* to the phenomenon, another dataset in which it is *not sensitive* and finally a theoretical model of our *expected signal*.

[1] 'Toy' is a common terminology for randomly sampled cases in Monte Carlo calculations. In our case, sampling representative candidates from the background distribution yields a set of 'toy candidates'.

Any data entry (after cleaning) in the sensitive dataset is a *candidate*. Each candidate is drawn from a distribution of the present background plus the expected signal contribution ($c = s + b$). Any entry in the non sensitive dataset is *background* only.

Goal Compute the value of a parameter (coupling constant) such that there is 95 % confidence that the combined hypothesis of signal and background sources are compatible with the background only hypothesis.

Condition Our experiment should be such that the data in some "channels" of our choice can be modeled by a Poisson distribution

$$P_{\text{Pois}}(k; \lambda) = \frac{\lambda^k e^{-\lambda}}{k!}.$$

Each such channel with an expectation value of λ counts has probability $P_{\text{Pois}}(k; \lambda)$ to measure k counts. Because the Poisson distribution (as written here) is a probability density function, multiple different channels can be combined to a "likelihood" for an experiment outcome by taking the product of each channel's Poisson probability

$$\mathcal{L}(\lambda) = \prod_i P_{i,\text{Pois}}(k; \lambda) = \prod_i \frac{\lambda_i^{k_i} e^{-\lambda_i}}{k_i!}.$$

Given a set of k_i recorded counts for all different channels i with expectation value λ_i the "likelihood" gives us the literal likelihood to record exactly that experimental outcome. Note that the parameter of the likelihood function is the mean λ and not the recorded data k! The likelihood function describes the likelihood for a **fixed set of data** (our real measured counts) for different parameters (our signal & background models, where the background model is constant as well).

In addition, the method described in the next section is valid under the assumption that our experiment did not have a statistically significant detection in the signal sensitive dataset compared to the background dataset.

12.2. Limit method - likelihood function $\mathcal{L}$

The likelihood function as described in the previous section is not optimal to compute a limit for the usage with different datasets as described before. [2] For that case we want to have some kind of a "test statistic" that relates the sensitive dataset with its seen candidates to the background dataset. For

[2] We could use $P_i(k_i; \lambda_i = s_i + b_i)$, but among other things a ratio is numerically more stable.

practical purposes we prefer to define a statistic, which is monotonically increasing in the number of candidates (see for example [124]). There are different choices possible, but the one we use is

$$\mathcal{L}(s,b) = \prod_i \frac{P_{\text{pois}}(c_i; s_i + b_i)}{P_{\text{pois}}(c_i; b_i)},$$

so the ratio of the signal plus background over the pure background hypothesis. The number c_i is the real number of measured **candidates**. So the numerator gives the probability to measure c_i counts in each channel i given the signal plus background hypothesis. On the other hand the denominator measures the probability to measure c_i counts in each channel i assuming only the background hypothesis.

> Note: For each channel i the ratio of probabilities itself is not strictly speaking a probability density function, because the integral
>
> $$\int_{-\infty}^{\infty} Q \, \mathrm{d}x = N \neq 1,$$
>
> with Q an arbitrary distribution. N can be interpreted as a hypothetical number of total number of counts measured in the experiment. A PDF requires this integral to be 1.
>
> As a result the full construct $\mathcal{L}$ of the product of these ratios is technically not a likelihood function either. It is usually referred to as an "extended likelihood function".
>
> For all practical purposes though we will continue to treat is as a likelihood function and call it $\mathcal{L}$ as usual.

Note the important fact that $\mathcal{L}$ really is only a function of our signal hypothesis s and our background model b. Each experimental outcome **has its own** $\mathcal{L}$. This is precisely why the likelihood function describes everything about an experimental outcome (at least if the signal and background models are reasonably understood) and thus different experiments can be combined by combining them in "likelihood space" (multiplying their $\mathcal{L}$ or adding $\ln \mathcal{L}$ values) to get a combined likelihood to compute a limit for.

Deriving a practical version of $\mathcal{L}$ The version of $\mathcal{L}$ presented above is still quite impractical to use and the ratio of exponentials of the Poisson distributions can be simplified significantly:

$$\begin{aligned}
\mathcal{L} &= \prod_i \frac{P(c_i, s_i + b_i)}{P(c_i, b_i)} = \prod_i \frac{\frac{(s_i+b_i)^{n_i}}{n_i!} e^{-(s_i+b_i)}}{\frac{b_i^{n_i}}{n_i!} e^{-b_i}} \\
&= \prod_i \frac{e^{-s_i}(s_i + b_i)^{c_i}}{b_i^{c_i}} = e^{-s_{\text{tot}}} \prod_i \frac{(s_i + b_i)^{c_i}}{b_i^{c_i}} \\
&= e^{-s_{\text{tot}}} \prod_i \left(1 + \frac{s_i}{b_i}\right)^{c_i}
\end{aligned}$$

This really is the heart of computing a limit with a number of s_{tot} expected events from the signal hypothesis (depending on the parameter to be studied, the coupling constant), c_i measured counts in each channel and s_i expected signal events and b_i expected background events in that channel.

As mentioned previously though, the choice of what a channel is, is completely up to us! One such choice might be binning the candidates in energy. However, there is one choice that is particularly simple and is often referred to as the "unbinned likelihood". Namely, we create channels in time such that each "time bin" is so short as to either have 0 entries (most channels) or 1 entry. This means we have a large number of channels, but because of our definition of $\mathcal{L}$ this does not matter. All channels with 0 candidates do not contribute to $\mathcal{L}$ (they are $\left(1 + \frac{s_i}{b_i}\right)^0 = 1$). As a result our expression of $\mathcal{L}$ simplifies further to:

$$\mathcal{L} = e^{-s_{\text{tot}}} \prod_i \left(1 + \frac{s_i}{b_i}\right)$$

where i is now all channels where a candidate is contained ($c_i = 1$).

12.3. Limit method - computing $\mathcal{L}$

Our simplified version of $\mathcal{L}$ using very short time bins now allows to explicitly compute the likelihood for a set of parameters. Let's now look at each of the constituents s_{tot}, s_i and b_i and discuss how they are computed. We will focus on the explicit case of an X-ray detector behind a telescope at CAST.

Here it is important to note that the signal hypothesis depends on the coupling constant we wish to compute a limit for, we will just call it g in the remainder (it may be $g_{a\gamma}$ or g_{ae} or any other coupling constant). This is the actual parameter of $\mathcal{L}$.

First of all the signal contribution in each channel s_i. It is effectively a number of counts that one would expect within the time window of the channel i. While this seems tricky given that we have not explicitly defined such a window we can:

- either assume our time interval to be infinitesimally small and give a signal rate (i.e. per second)
- or make use of the neat property that our expression only contains **the ratio** of s_i and b_i. What this means is that we can choose our units ourselves, as long as we use the same units for s_i as for b_i!

We will use the second case and scale each candidate's signal and background contribution to the total tracking time (signal sensitive dataset length). Each parameter with a subscript i is the corresponding value that the candidate has we are currently looking at (e.g. E_i is the energy of the recorded candidate

i used to compute the expected signal).

$$s_i(g^2) = f(g^2, E_i) \cdot A \cdot t \cdot P_{a\to\gamma}(g_{a\gamma}^2) \cdot \varepsilon(E_i) \cdot r(x_i, y_i) \tag{12.1}$$

where:

- $f(g, E_i)$ is the axion flux at energy E_i in units of $\mathrm{keV}^{-1}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$ as a function of g^2, sec. 12.8.2,
- A is the area of the magnet bore in cm^2, sec. 12.8.1,
- t is the tracking time in s, also sec. 12.8.1,
- $P_{a\to\gamma}$ is the conversion probability of the axion converting into a photon computed via

  $$P_{a\to\gamma}(g_{a\gamma}^2) = \left(\frac{g_{a\gamma} B L}{2}\right)^2$$

  written in *natural units* (meaning we need to convert B and L into values expressed in powers of eV, as discussed in sec. 3.3), sec. 12.8.3,
- $\varepsilon(E_i)$ is the combined detection efficiency, i.e. the combination of X-ray telescope effective area, the transparency of the detector window and the absorption probability of an X-ray in the gas, sec. 12.8.4,
- $r(x_i, y_i)$ is the expected amount of flux from solar axions after it is focused by the X-ray telescope in the readout plane of the detector at the candidate's position (x_i, y_i) (this requires a raytracing model). It should be expressed as a fractional value in units of cm^{-2}. See sec. 12.8.6.

As a result the units of s_i are then given in $\mathrm{keV}^{-1}\,\mathrm{cm}^{-2}$ with the tracking time integrated out. If one computes a limit for $g_{a\gamma}$ then f and P both depend on the coupling of interest, making s_i a function of $g_{a\gamma}^4$. In case of e.g. an axion-electron coupling limit g_{ae}, the conversion probability can be treated as constant (with a fixed $g_{a\gamma}$).

Secondly, the background hypothesis b_i for each channel. Its value depends on whether we assume a constant background model, an energy dependent one or even an energy plus position dependent model. In either case the main point is to evaluate that background model at the position (x_i, y_i) of the candidate and energy E_i of the candidate. The value should then be scaled to the same units of as s_i, namely $\mathrm{keV}^{-1}\,\mathrm{cm}^{-2}$. Depending on how the model is defined this might just be a multiplication by the total tracking time in seconds. We discuss this in detail in sec. 12.8.8.

The final piece is the total signal s_{tot}, corresponding to the total number of counts expected from our signal hypothesis for the given dataset. This is nothing else as the integration of s_i over the entire energy range and detection area. However, because s_i implies the signal for candidate i, we write $s(E, x, y)$ to mean the equivalent signal as if we had a candidate at (E, x, y)

$$s_{\text{tot}} = \int_0^{E_{\text{max}}} \int_A s(E, x, y)\, \mathrm{d}E\, \mathrm{d}x\, \mathrm{d}y$$

where A simply implies integrating the full area in which (x, y) is defined. The axion flux is bounded within a region much smaller than the active detection area and hence all contributions outside are 0.

12.4. Limit method - computing a limit

With the above we are now able to evaluate $\mathcal{L}$ for a set of candidates $c_i(E_i, x_i, y_i)$. As mentioned before it is important to realize that $\mathcal{L}$ is a function of the coupling constant g, $\mathcal{L}(g)$ with all other parameters effectively constant in the context of "one experiment". g is a placeholder for the parameter, in which $\mathcal{L}$ is linear, i.e. g_{ae}^2 for axion-electron or $g_{a\gamma}^4$ and β_γ^4 for axion-photon and chameleon, respectively.

With this in mind the 'limit' is defined as the 95-th percentile of $\mathcal{L}(g)$ *within the physical region of* g. The region $g < 0$ is explicitly ignored, as a coupling constant cannot be negative! This can be rigorously justified in Bayesian statistics by saying the prior $\pi(\mathrm{g})$ is 0 for $g < 0$.

We can define the limit implicitly as [3]

$$0.95 = \frac{\int_0^{g'} \mathcal{L}(g)\, \mathrm{dg}}{\int_0^{\infty} \mathcal{L}(g)\, \mathrm{dg}} \tag{12.2}$$

In practice the integral cannot be evaluated until infinity. Fortunately, our choice of $\mathcal{L}$ in the first place means that the function converges to 0 quickly for large values of g. Therefore, we only need to compute values to a "large enough" value of g to get the shape of $\mathcal{L}(g)$. From there we can use any numerical approach (via an empirical cumulative distribution function for example) to determine the coupling constant g' that corresponds to the 95-th percentile of $\mathcal{L}(g)$.

In an intuitive sense the limit means the following: 95 % of all coupling constants that reproduce the data we measured – given our signal and background hypotheses – are smaller than g'.

Fig. 12.1 shows an example of a likelihood function of some coupling constant. The blue area is the lower 95 % of the parameter space and the red area is the upper 5 %. Therefore, the limit in this particular set of toy candidates is at the intersection of the two colors.

12.5. Limit method - toy candidate sets and expected limits

Assuming a constant background over some chip area with only an energy dependence, the background hypothesis can be used to draw toy candidates that can be used in place for the real candidates to compute limits. In this situation the background hypothesis can be modeled as follows:

[3] This equation essentially computes the confidence level at CL = 95 % ≡ 0.95 = 1 − 0.05 = 1 − α. In the equation we already removed the prior and therefore adjusted the integration range.

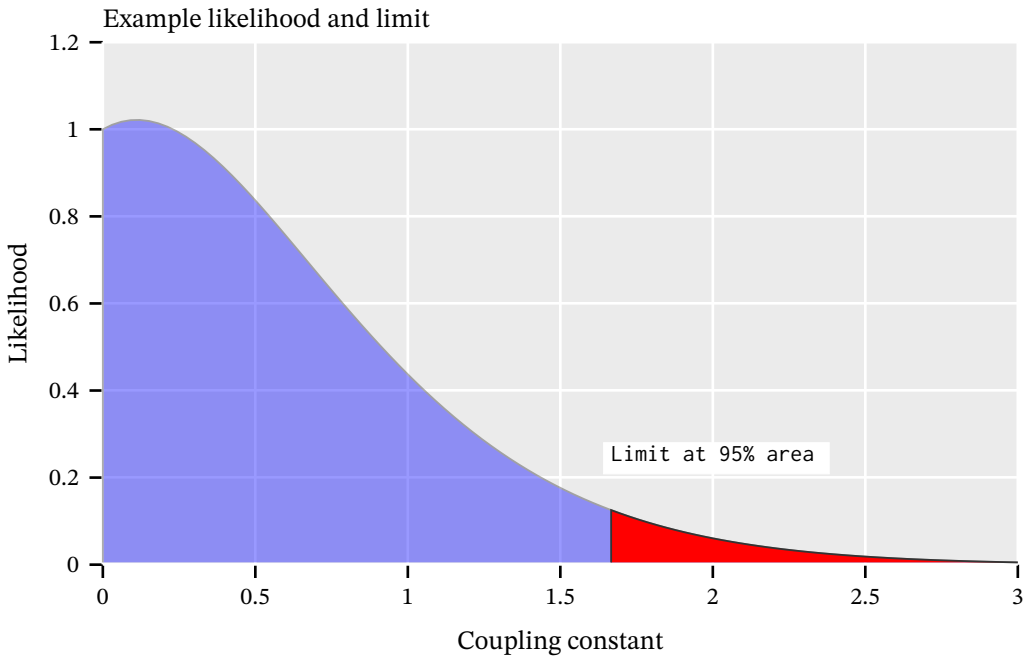


Figure 12.1.: Example likelihood function for a set of toy candidates. Blue is the lower 95-th percentile of the integral over the likelihood function and red the upper 5-th. The limit is at the intersection.

$$B = \{P_{\text{Pois}}(k; \lambda = b_i) \mid \text{for all energy bins } E_i\},$$

that is, the background is the set of all energy bins E_i, where each bin content is described by a Poisson distribution with a mean and expectation value of $\lambda = b_i$ counts.

To compute a set of toy candidates then, we simply iterate over all energy bins and draw a number from each Poisson distribution. This is the number of candidates in that bin for the toy. Given that we assumed a constant background over the chip area, we finally need to draw the (x_i, y_i) positions for each toy candidate from a uniform distribution. [4]

These sets of toy candidates can be used to compute an "expected limit". The term expected limit is usually understood to mean the median of many limits computed based on representative toy candidate sets. If L_{t_i} is the limit of the toy candidate set t_i, the expected limit $\langle L \rangle$ is defined as

$$\langle L \rangle = \text{median}(\{L_{t_i}\})$$

If the number of toy candidate sets is large enough, the expected limit should prove accurate. The real limit will then be below or above with 50 % chance each.

[4] If one considers a position independent likelihood function, there is no need to sample positions of course.

12.6. Limit method - extending $\mathcal{L}$ for systematics

The aforementioned likelihood ratio assumes perfect knowledge of the inputs for the signal and background hypotheses. In practice neither of these is known perfectly though. Each input typically has associated a small systematic uncertainty (e.g. the width of the detector window is only known up to N nanometers, the pressure in the chamber only up to M millibar, magnet length only up to C centimeters etc.). These all affect the "real" numbers one should actually calculate with. So how does one treat these uncertainties?

The basic starting point is realizing that the values we use are our "best guess" of the real value. Usually it is a reasonable approximation that the real value will likely be within some standard deviation around that best guess, following a normal distribution. Further, it is a good idea to identify all systematic uncertainties and classify them by which aspect of s_i, b_i or (x_i, y_i) they affect (amount of signal, background or the position [5]). Another reasonable assumption is to combine different uncertainties of the same type by

$$\Delta x = \sqrt{\sum_{i=1}^{N} \Delta x_i^2},$$

i.e. computing the euclidean radius in N dimensions, for N uncertainties of the same type.

The above explanation can be followed to encode these uncertainties into the limit calculation. For a value corresponding to our "best guess" we want to recover the likelihood function $\mathcal{L}$ from before. The further we get away from our "best guess", the more the likelihood function should be "penalized", meaning the actual likelihood of that parameter given our data should be **lower**. The initial likelihood $\mathcal{L}$ will be modified by multiplying with additional normal distributions, one for each uncertainty (4 in total in our case, signal, background, and two position uncertainties). Each adds an additional parameter, a 'nuisance parameter'.

To illustrate the details, let's look at the case of adding a single nuisance parameter. In particular we'll look at the nuisance parameter for the signal as it requires more care. The idea is to express our uncertainty of a parameter – in this case the signal – by introducing an additional parameter s_i'. In contrast to s_i it describes a possible other value of s_i due to our systematic uncertainty. For simplicity we rewrite our likelihood $\mathcal{L}$ as $\mathcal{L}'(s_i, s_i', b_i)$:

$$\mathcal{L}' = e^{-s_{\text{tot}}'} \prod_i \left(1 + \frac{s_i'}{b_i}\right) \cdot \exp\left[-\frac{1}{2}\left(\frac{s_i' - s_i}{\sigma_s'}\right)^2\right]$$

where s_i' takes the place of the s_i. The added gaussian then provides a penalty for any deviation from s_i. The standard deviation of the gaussian σ_s' is the actual systematic uncertainty on our parameter s_i in units of s_i.

[5] For different likelihood functions other parameters may be affected.

This form of adding a secondary parameter s_i' of the same units as s_i is not the most practical, but maybe provides the best explanation as to how the name 'penalty term' arises for the added gaussian. If $s_i = s_i'$ then the exponential term is 1 meaning the likelihood remains unchanged. For any other value the exponential is < 1, decreasing the likelihood $\mathcal{L}'$.

By a change of variables we can replace the "unitful" parameter s_i' by a unitless number ϑ_s. We would like the exponential to be $\exp(-\vartheta_s^2/(2\sigma_s^2))$ to only express deviation from our best guess s_i. $\vartheta_s = 0$ means no deviation and $|\vartheta_s| = 1$ implies $s_i = -s_i'$. Note that the standard deviation of this is now σ_s and **not** σ_s' as seen in the expression above. This σ_s corresponds to our systematic uncertainty on the signal as a percentage.

To arrive at this expression:

$$\begin{aligned}
\frac{s_i' - s_i}{\sigma_s'} &= \frac{\vartheta_s}{\sigma_s} \\
\Rightarrow s_i' &= \frac{\sigma_s'}{\sigma_s}\vartheta_s + s_i \\
\text{with } s_i &= \frac{\sigma_s'}{\sigma_s} \\
s_i' &= s_i + s_i\vartheta_s \\
\Rightarrow s_i' &= s_i(1 + \vartheta_s)
\end{aligned}$$

where we made use of the fact that the two standard deviations are related by the signal s_i (which can be seen by defining ϑ_s as the normalized difference $\vartheta_s = \frac{s_i' - s_i}{s_i}$).

This results in the following final (single nuisance parameter) likelihood $\mathcal{L}'$:

$$\mathcal{L}' = e^{-s_{\text{tot}}'} \prod_i \left(1 + \frac{s_i'}{b_i}\right) \cdot \exp\left[-\frac{1}{2}\left(\frac{\vartheta_s}{\sigma_s}\right)^2\right]$$

where $s_i' = s_i(1 + \vartheta_s)$ and similarly $s_{\text{tot}}' = s_{\text{tot}}(1 + \vartheta_s)$ (the latter just follows because $1 + \vartheta_s$ is a constant under the different channels i).

The same approach is used to encode the background systematic uncertainty. The position uncertainty is generally handled the same, but the x and y coordinates are treated separately.

As shown in eq. 12.1 the signal s_i actually depends on the positions (x_i, y_i) of each candidate via the raytracing image r. With this we can introduce the nuisance parameters by replacing r by an r' such that

$$r' \mapsto r(x_i - x_i', y_i - y_i')$$

which effectively moves the center position by (x_i', y_i'). In addition we need to add penalty terms for each of these introduced parameters:

$$\mathcal{L}' = \exp[-s] \cdot \prod_i \left(1 + \frac{s_i'}{b_i}\right) \cdot \exp\left[-\left(\frac{x_i - x_i'}{\sqrt{2}\sigma}\right)^2\right] \cdot \exp\left[-\left(\frac{y_i - y_i'}{\sqrt{2}\sigma}\right)^2\right]$$

where s_i' is now the modification from above using r' instead of r. Now we perform the same substitution as we do for ϑ_b and ϑ_s to arrive at:

$$\mathcal{L}' = \exp[-s] \cdot \prod_i \left(1 + \frac{s_i'}{b_i}\right) \cdot \exp\left[-\left(\frac{\vartheta_x}{\sqrt{2}\sigma_x}\right)^2\right] \cdot \exp\left[-\left(\frac{\vartheta_y}{\sqrt{2}\sigma_y}\right)^2\right]$$

The substitution for r' means the following for the parameters:

$$r' = r\left(x(1 + \vartheta_x), y(1 + \vartheta_y)\right)$$

where essentially a deviation of $|\vartheta| = 1$ means we move the center of the axion image to the edge of the chip.

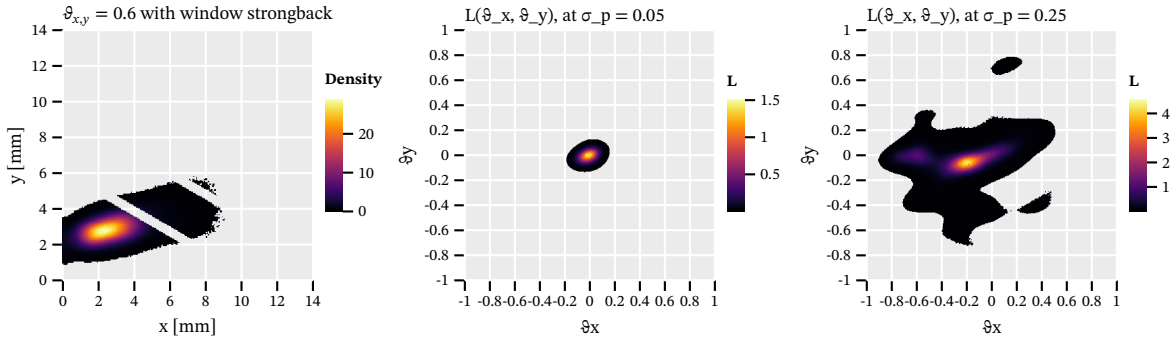
Putting all these four nuisance parameters together we have

$$\mathcal{L}' = \left(\prod_i \frac{P_{\text{pois}}(n_i; s_i + b_i)}{P_{\text{pois}}(n_i; b_i)}\right) \cdot \mathcal{N}(\vartheta_s, \sigma_s) \cdot \mathcal{N}(\vartheta_b, \sigma_b) \cdot \mathcal{N}(\vartheta_x, \sigma_x) \cdot \mathcal{N}(\vartheta_y, \sigma_y) \tag{12.3}$$

$$\mathcal{L}'(\mathrm{g}, \vartheta_s, \vartheta_b, \vartheta_x, \vartheta_y) = e^{-s'_{\text{tot}}} \prod_i \left(1 + \frac{s_i''}{b_i'}\right) \cdot \exp\left[-\frac{1}{2}\left(\frac{\vartheta_s}{\sigma_s}\right)^2 - \frac{1}{2}\left(\frac{\vartheta_b}{\sigma_b}\right)^2 - \frac{1}{2}\left(\frac{\vartheta_x}{\sigma_x}\right)^2 - \frac{1}{2}\left(\frac{\vartheta_y}{\sigma_y}\right)^2\right] \tag{12.4}$$

where here the doubly primed s_i'' refers to modification for the signal nuisance parameter as well as for the position uncertainty via r'.

An example of the impact of the nuisance parameters on the likelihood space as well as on the parameters (s, b, x, y) is shown in fig. 12.2. First, fig. 12.2(a) shows how the axion image moves when $\vartheta_{x,y}$ change, in this example $\vartheta_{x,y} = 0.6$ moves the image center to the bottom left ($\vartheta_{x,y} = 1$ would move the center into the corner). Note that the window strongback is not tied to the axion image, but remains fixed (the cut out diagonal lines). Fig. 12.2(b) and 12.2(c) show the impact of the nuisance parameters on the likelihood space. The larger the standard deviation $\sigma_{x,y}$ is, the more of the $\vartheta_{x,y}$ space contributes meaningfully to $\mathcal{L}_M$. In the former example – a realistic uncertainty – only small regions around the center are allowed to contribute. Regions further outside receive too large of a penalty. However, at large uncertainties significant regions of the parameter space play an important role. Given that each point on the figures 12.2(b) and 12.2(c) describes one axion image like 12.2(a), brighter regions imply positions where the axion image is moved to parts that provide a larger s/b in the center portion of the axion image, while still only having a small enough penalty. For the realistic uncertainty, $\sigma = 0.05$, roughly the inner $-0.1 < \vartheta < 0.1$ space contributes. This corresponds to a range of $-0.7\,\text{mm} < x < 0.7\,\text{mm}$ around the center in fig. 12.2(a).



(a) Axion image at $\vartheta_{x,y} = 0.6$ (b) $\sigma_{x,y} = 0.05$ (c) $\sigma_{x,y} = 0.25$

Figure 12.2.: (a): Impact of the position nuisance parameter on the axion image. A value of $\vartheta_{x,y} = 0.6$ is shown, moving the center of the image to the bottom left corner. (b) shows the impact on the likelihood itself for varying $\vartheta_{x,y}$ values given a standard deviation of $\sigma_{x,y} = 0.05$. Small variations of the position still yield contributions to $\mathcal{L}_M$. (c) the same for $\sigma_{x,y} = 0.25$. At this value large regions of the $\vartheta_{x,y}$ parameter space contribute to $\mathcal{L}_M$, generally regions of larger s/b.

12.7. Limit method - evaluating $\mathcal{L}$ with nuisance parameters

The likelihood function we started with $\mathcal{L}$ was only a function of the coupling constant g we want to compute a limit for. With the inclusion of the four nuisance parameters however, $\mathcal{L}'$ is now a function of 5 parameters, $\mathcal{L}'(g, \vartheta_s, \vartheta_b, \vartheta_x, \vartheta_y)$. Following our definition of a limit via a fixed percentile of the integral over the coupling constant, eq. 12.2, leads to a problem for $\mathcal{L}'$. If anything, one could define a contour describing the 95-th percentile of the "integral volume", but this would lead to infinitely many values of g that describe said contour.

As a result, to still define a sane limit value, the concept of the marginal likelihood function $\mathcal{L}'_M$ is introduced. The idea is to integrate out the nuisance parameters

$$\mathcal{L}'_M(g) = \iiiint_{-\infty}^{\infty} \mathcal{L}'(g, \vartheta_s, \vartheta_b, \vartheta_x, \vartheta_y)\, \mathrm{d}\vartheta_s\, \mathrm{d}\vartheta_b\, \mathrm{d}\vartheta_x\, \mathrm{d}\vartheta_y. \tag{12.5}$$

Depending on the exact definition of $\mathcal{L}'$ in use, these integrals may be analytically computable. In many cases however they are not and numerical techniques to evaluate the integral must be utilized.

Aside from the technical aspects about how to evaluate $\mathcal{L}'_M(g)$ at a specific g, the limit calculation continues exactly as for the case without nuisance parameters once $\mathcal{L}'_M(g)$ is defined as such.

Practical calculation of $\mathcal{L}'_M(g)$ in our case In case of our explicit likelihood function eq. 12.3 there is already one particular case that makes the marginal likelihood not analytically integrable because the $b'_i = b_i(1 + \vartheta_b)$ term introduces a singularity for $\vartheta_b = -1$. For practical purposes this is not too relevant, as values approaching $\vartheta_b = -1$ would imply having zero background and within a reasonable systematic uncertainty the penalty term makes contributions in this limit so small such that this area does not physically contribute to the integral.

Using standard numerical integration routines (simpson, adaptive Gauss-Kronrod etc.) are all too expensive to compute such a four-fold integration under the context of computing many toy limits for an expected limit. For this reason Monte Carlo approaches are used, in particular the Metropolis-Hastings [152, 103] (MH) Markov Chain Monte Carlo (MCMC). The basic idea of general Monte Carlo integration routines is to evaluate the function at random points and computing the integral based on the function evaluation at these points (by scaling the evaluations correctly). Unless the function is very 'spiky' in the integration space, Monte Carlo approaches provide good accuracy at a fraction of the computational effort as normal numerical algorithms even in higher dimensions. However, we can do better than relying on fully random points in the integration space. The Metropolis-Hastings algorithm tries to evaluate the function more often in those points where the contributions are large. The basic idea is the following:

Pick a random point in the integration space as a starting point p_0. Next, pick another random point p_1 within the vicinity of p_0. If the function f evaluates to a larger value at p_1, accept it as the new current position. If it is smaller, accept it with a probability of $\frac{f(p_i)}{f(p_{i-1})}$ (i.e. if the new value is close to the old one we accept it with a high probability and if the new one is much lower accept it rarely). This guarantees to pick values inching closer to the most contributing areas of the integral in the integration space, while still allowing to get out of local maxima due to the random acceptance of "worse" positions. However, this also implies that regions of constant $\mathcal{L}$ (regions where the values are close to 0, but also generally 'flat' regions) produce a pure random walk from the algorithm, because $\frac{f(p_i)}{f(p_{i-1})} \approx 1$ in those regions. This needs to be taken into account.

If a new point is accepted and becomes the current position, the "chain" of points is extended (hence "Markov Chain"). If a point is rejected, extend the chain by duplicating the last point. By creating a chain of reasonable length, the integration space is sampled well. Because the initial point is completely random (up to some possible prior) the first N links of the chain are in a region of low interest (and depending on the interpretation of the chain "wrong"). For that reason one defines a cutoff N_b of the first elements that are thrown away as "burn-in" before using the chain to evaluate the integral or parameters.

In addition it can be valuable to not only start a single Markov Chain from one random point, but instead start *multiple* chains from different points in the integration space. This increases the chance to cover different regions of interest even in the presence of multiple peaks separated too far away to likely "jump over" via the probabilistic acceptance. As such it reduces bias from the starting sampling.

To summarize the algorithm:

1. let $\vec{p}$ be a random vector in the integration space and $f(\vec{p})$ the function to evaluate,
2. pick new point $\vec{p}'$ in vicinity of $\vec{p}$,
3. sample from random uniform in $[0, 1]$: u,
4. accept $\vec{p}'$ if $u < \frac{f(\vec{p}')}{f(\vec{p})}$, add $\vec{p}'$ to chain and iterate (if $f(\vec{p}') > f(\vec{p})$ every new link accepted!). If rejected, add $\vec{p}$ again,
5. generate a long enough chain to sample the integration space well,
6. throw away first N elements as "burn in",
7. generate multiple chains to be less dependent on starting position.

Applied to eq. 12.5, we obtain $\mathcal{L}_M(g)$ by computing the histogram of all sampled g values, which are one component of the vector $\vec{p}$. More on that in sec. 12.10.

12.8. Likelihood ingredients in detail

To reiterate, the likelihood function we finally evaluate using MCMC, with explicit dependency on the coupling constant we intend to (mainly) consider – the axion-electron coupling g_{ae} – can be written as

$$\mathcal{L}'_M(g_{ae}^2) = \iiiint_{-\infty}^{\infty} e^{-s'_{\text{tot}}(g_{ae}^2)} \cdot \prod_i \left(1 + \frac{s''_i(g_{ae}^2)}{b'_i}\right) \cdot \exp\left[-\frac{\vartheta_b^2}{2\sigma_b^2} - \frac{\vartheta_s^2}{2\sigma_s^2} - \frac{\vartheta_x^2}{2\sigma_x^2} - \frac{\vartheta_y^2}{2\sigma_y^2}\right] \mathrm{d}\vartheta_b \,\mathrm{d}\vartheta_s \,\mathrm{d}\vartheta_x \,\mathrm{d}\vartheta_y,$$

where i runs over all candidates. We alluded to the general make up of both the signal terms s_{tot} and s_i as well as the background b_i in sec. 12.3. Let us now look at what goes into each of these explicitly and how they are calculated, starting with each of the signal contributions in

$$s_i(g_{ae}^2) = f(g_{ae}^2, E_i) \cdot A \cdot t \cdot P_{a\to\gamma}(g_{a\gamma}^2) \cdot \varepsilon(E_i) \cdot r(x_i, y_i),$$

sec. 12.8.1 to sec. 12.8.7 and the background in sec. 12.8.8. Finally, sec. 12.8.9 explains how we sample toy candidate sets.

12.8.1. Magnet bore and solar tracking time - A, t

Starting with the simplest inputs to the signal, the magnet bore area and the solar tracking time. The CAST magnet has a bore diameter of $d_{\text{bore}} = 43\,\text{mm}$, as introduced in sec. 4.1. The relevant area for the solar axion flux is the entire magnet bore, because the X-ray telescope covers the full area. As such, A is a constant of:

$$A = \pi(21.5\,\text{mm})^2 = 1452.2\,\text{mm}^2.$$

The time of interest is the total solar tracking duration, in which the detector was sensitive (i.e. removing the dead time due to readout). As given in the CAST data taking overview, sec. 9.6, the amount of active solar tracking time is

$$t = 160.38\,\text{h}.$$

12.8.2. Solar axion flux - $f(g, E_i)$

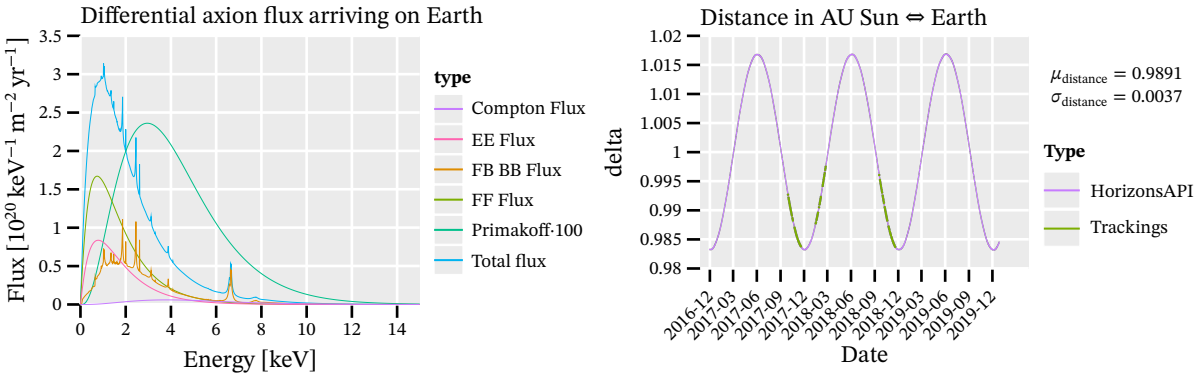
The solar axion flux is based on the calculations by J. Redondo [178] as already introduced in sec. 3.4. The $f(g^2, E_i)$ term of the signal refers to the differential solar axion flux. The flux, fig. 12.3(a), is computed for a specific axion model and coupling constant, in this case $g_{\text{ref}} = g_{ae} = 10^{-13}$ and $g_{a\gamma} = 10^{-12}\,\text{GeV}^{-1}$. As the flux scales by the coupling constant squared, it is rescaled to a new coupling constant g_{ae}^2 by

$$f(g_{ae}^2, E_i) = f(g_{ae,\text{ref}}^2, E_i) \cdot \frac{g_{ae}^2}{g_{\text{ref},ae}^2}.$$

$g_{a\gamma}$ is kept constant. At this ratio of the two coupling constants, the axion-photon flux is negligible.

The shown differential flux is computed using a Sun to Earth distance of $d_{S\Leftrightarrow E} = 0.989$ AU due to the times of the year in which solar trackings were taken at CAST. Fig. 12.3(b) shows the distance between Sun and Earth during the entire data taking period, with the solar trackings marked in green. The data for the distance is obtained using the JPL Horizons API [121].

The code used to calculate the differential flux, [159] [6], can also be used to compute the flux for other axion models, for example a pure axion-photon coupling model.



(a) Solar axion flux

(b) Distance Sun-Earth

Figure 12.3.: (a): Differential solar axion flux assuming a distance to the Sun of 0.989 AU based on (b).

[6] The code was mainly developed by Johanna von Oy under my supervision. I only contributed minor feature additions and refactoring, as well as performance related improvements.

12.8.3. Conversion probability - $P_{a\gamma}(g_{a\gamma}^2)$

The conversion probability of the arriving axions is simply a constant factor, depending on $g_{a\gamma}$, see section 3.3 for the derivation from the general formula. The simplified expression for coherent conversion [7] in a constant magnetic field [8] is

$$P(g_{a\gamma}^2, B, L) = \left(\frac{g_{a\gamma} \cdot B \cdot L}{2} \right)^2$$

where the relevant numbers for the CAST magnet are:

$$\begin{aligned} B &= 8.8\,\text{T} && \mapsto B_{\text{natural}} = 1719.1\,\text{eV}^2 \\ L &= 9.26\,\text{m} && \mapsto L_{\text{natural}} = 4.692\,72 \times 10^7\,\text{eV}^{-1}. \end{aligned}$$

The magnetic field is taken from the CAST slow control log files and matches the values used in the paper of CAST CAPP [3] (in contrast to some older papers which assumed 9 T, based on when the magnet was still intended to be run at above 13 000 A).

Assuming a fixed axion-photon coupling of $g_{a\gamma} = 10^{-12}\,\text{GeV}^{-1}$ the conversion probability comes out to:

$$\begin{aligned} P(g_{a\gamma}^2, B, L) &= \left(\frac{g_{a\gamma} \cdot B \cdot L}{2} \right)^2 \\ &= \left(\frac{10^{-12}\,\text{GeV}^{-1} \cdot 1719.1\,\text{eV}^2 \cdot 4.693 \times 10^7\,\text{eV}^{-1}}{2} \right)^2 \\ &= 1.627 \times 10^{-21} \end{aligned}$$

12.8.4. Detection efficiency - $\varepsilon(E_i)$

The detection efficiency $\varepsilon(E_i)$ includes multiple aspects of the full setup. It can be further decomposed into the telescope efficiency, window transparency, gas absorption, software efficiency of the classifier and veto efficiency,

$$\varepsilon(E_i) = \varepsilon_{\text{telescope}}(E_i) \cdot \varepsilon_{\text{window}}(E_i) \cdot \varepsilon_{\text{gas}}(E_i) \cdot \varepsilon_{\text{software eff.}} \cdot \varepsilon_{\text{veto eff.}}.$$

[7] Note that to calculate limits for larger axion masses the sinc term of eq. 3.2 needs to be included.

[8] Also note that in a perfect analysis one would compute the conversion in a realistic magnetic field, as the field strength is not perfectly homogeneous. That would require a very precise field map of the magnet. In addition, the calculations for axion conversions in inhomogeneous magnetic fields is significantly more complicated. As far as I understand it requires essentially a "path integral like" approach of all possible paths through the magnet, where each path sees different, varying field strengths. Due to the small size of the LHC dipole prototype magnet and general stringent requirements for homogeneity this is not done for this analysis. However, likely for future (Baby)IAXO analyses this will be necessary.

The first three are energy dependent and the latter two constants, but dependent on the classifier and veto setup for which we compute limits.

Telescope efficiency - $\varepsilon_{\text{telescope}}(E_i)$

The X-ray telescope further has a direct impact not only on the shape of the axion signal on the readout, but also the total number of X-rays transmitted. The effective transmission of an X-ray telescope is significantly lower than in the optical range. This is typically quoted using the term "effective area". In section 4.1.1 the effective area of the two X-ray optics used at CAST is shown. The term effective area refers to the equivalent area a perfect X-ray telescope (100 % transmission) would cover. As such, the real efficiency ε_{tel} can be computed by the ratio of the effective area A_{eff} and the total area of the optic A_{tel} exposed to light.

$$\varepsilon_{\text{tel}}(E) = \frac{A_{\text{eff}}(E)}{A_{\text{tel}}}$$

where the effective area A_{eff} depends on the energy. [9] In case of CAST the relevant total area is not actually the cross-sectional area of the optic itself, but rather the exposed area due to the diameter of the magnet coldbore. With a coldbore diameter of $d_{\text{bore}} = 43\,\text{mm}$ the effective area can be converted to ε_{tel}.

The resulting effective area is shown in fig. 12.4 in the next section together with the window transmission and gas absorption.

> Note: all publicly available effective areas for the LLNL telescope, meaning [120] and [28], are either inapplicable, outdated or unfortunately wrong. Jaime Ruz sent me the simulation results used for the CAST Nature paper [64], which include the effective area. These numbers are used in the figure below and our limit calculation.

Window transmission and argon gas absorption - $\varepsilon_{\text{window}}(E_i), \varepsilon_{\text{gas}}(E_i)$

The detector entrance window is the next point affecting the possible signal to be detected. The windows, as explained in section 6.9 are made from 300 nm thick silicon nitride with a 20 nm thick aluminium coating. Its transmission is very good down to about 1 keV below which it also starts to degrade rapidly.

[9] Note that ε_{tel} here is the average effective efficiency of the full telescope and *not* the reflectivity of a single shell. As a Wolter I optic requires two reflections ε_{tel} is equivalent to the reflectivity squared R^2. Individual reflectivities of shells are further complicated by the fact that different shells receive parallel light under different angles, which means the reflectivity varies between shells. Therefore, this is a measure for the average efficiency.

While the window also has four 500 μm thick strongbacks which in total occlude about 22.2 % of the center region, these are *not* taken into account into the combined detection efficiency. Instead they are handled together with the axion image $r(x_i, y_i)$ in sec. 12.8.6.

Software efficiency and veto efficiency - $\varepsilon_{\text{software eff.}} \cdot \varepsilon_{\text{veto eff.}}$

The software efficiency $\varepsilon_{\text{software eff.}}$ of course depends on the specific setting which is used. Its value will range from somewhere between 80 to 97 %. The veto efficiencies in principle can also vary significantly depending on the choice of parameters (e.g. whether the 'line veto' uses an eccentricity cutoff or not), but as explained in sec. 11.5.5 the septem and line vetoes are just considered as either yes or no. The FADC veto has also been fixed to a 1$^{\text{st}}$ to 99$^{\text{th}}$ percentile cut on the signal rise time, see sec. 11.5.2.

As such the relevant veto efficiencies are:

$$\begin{aligned} \varepsilon_{\text{FADC}} &= 98\,\% \\ \varepsilon_{\text{septem}} &= 83.11\,\% \\ \varepsilon_{\text{line}} &= 85.39\,\% \\ \varepsilon_{\text{septem+line}} &= 78.63\,\% \end{aligned}$$

where the last one corresponds to using both the septem and the line veto at the same time. Considering for example the case of using these vetoes together with a software efficiency of 80 % we see that the combined efficiency is already only about 61.6 %, which is an extreme loss in sensitivity.

Combined detection efficiency - $\varepsilon(E_i)$

The previous sections cover aspects which affect the detection efficiency of the detector and thus impact the amount of signal available. Combined they yield a detection efficiency as shown in fig. 12.4. As can be seen, the combined detection efficiency maxes out at about $\sim$ 46 % around 1.5 keV without taking into account the software and veto efficiencies. If one combines this with using all vetoes at a software efficiency of 80 %, the total detection efficiency of the detector would peak at only 28.4 % at that energy.

12.8.5. Average absorption depth of X-rays

In order to compute a realistic axion image based on raytracing, the plane at which to compute the image needs to be known, as the focal spot size changes significantly depending on the distance to

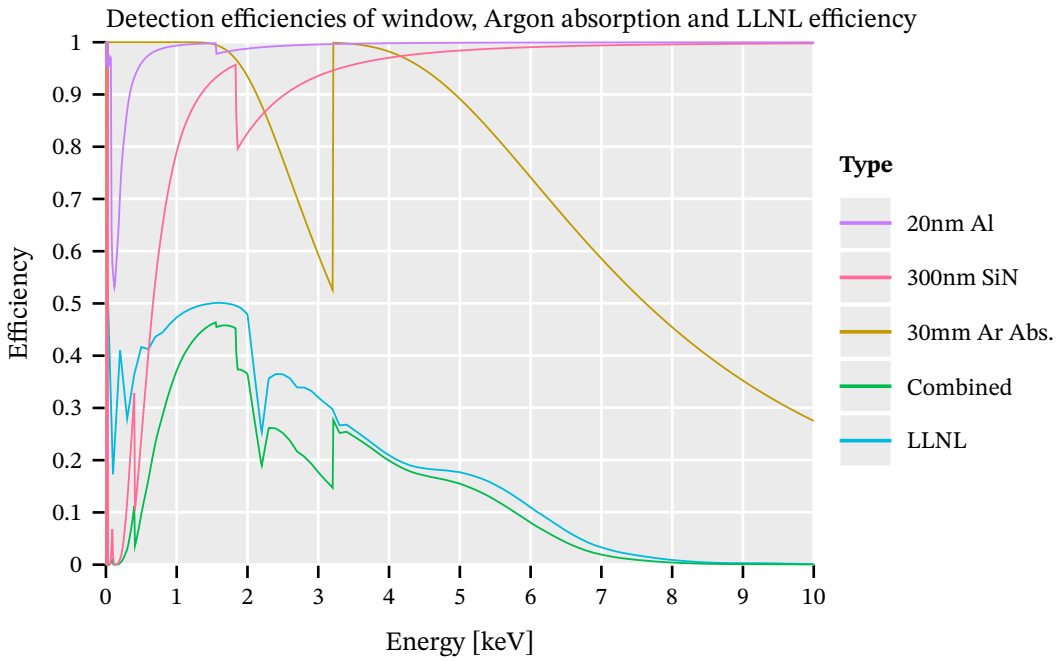


Figure 12.4.: The combined detection efficiency of the detector, taking into account the telescope efficiency via the effective area, the window absorption probability and the absorption probability in the detector gas.

the focal point of the X-ray optics. The beamline behind the telescope is designed such that the focal spot is 1 cm behind the entrance window. [10]

This is of particular importance for a gaseous detector, as the raytracing only makes sense up to the generation of a photoelectron, after which the produced primary electrons undergo diffusion. Therefore, one needs to compute the typical absorption depth of X-rays in the relevant energy ranges for the used gas mixture of the detector. This is easiest done based on a Monte Carlo simulation taking into account the incoming X-ray flux distribution (given the solar axion flux we consider) $f(E)$, the telescope effective area $\varepsilon_{\text{LLNL}}(E)$ and window transmission, $\varepsilon_{\text{Si}_3\text{N}_4}(E), \varepsilon_{\text{Al}}(E)$,

$$I(E) = f(E) \cdot \varepsilon_{\text{LLNL}}(E) \cdot \varepsilon_{\text{Si}_3\text{N}_4}(E) \cdot \varepsilon_{\text{Al}}(E).$$

$I(E)$ yields the correct energy distribution of expected signal X-rays. For each sampled X-ray we can then draw a conversion point based on the attenuation length and the Beer-Lambert law for its energy introduced in sec. 5.1.1. Computing the median of all conversion points is then an estimator for the point at which to compute the axion image.

Performing this calculation leads to a median conversion point of $\langle d \rangle = 0.2928\,\text{cm}$ behind the detector window, with a standard deviation of 0.4247 cm due to a long tail from higher energy X-rays. It may be worthwhile to perform this calculation for distinct energies to then compute different

[10] To my knowledge there exists no technical design documentation about how the beamline was designed exactly. Jaime Ruz, who was in charge of the LLNL telescope installation, told me this is what he aligned to.

axion images for different energies with each their own effective 'depth' behind the window, however for the time being we do not.

For the calculation of these numbers, see appendix Q.

12.8.6. Raytracing axion image - $r(x_i, y_i)$

The axion image is computed based on a raytracing Monte Carlo simulation, using TrAXer [193], written as part of this thesis. Appendix R contains an introduction to raytracing techniques, details about the LLNL telescope, verification of the raytracing results using PANTER measurements of the real telescope and details about the calculation of the axion image.

Fig. 12.5 shows the image, computed for a Sun-Earth distance of 0.989 AU and a distance of 0.2928 cm behind the detector window. So it is 0.7072 cm in front of the focal point. Hence, the image is very slightly asymmetric along the long axis.

Instead of using the raytracing image to fully characterize the axion flux including efficiency losses, we *only* use it to define the spatial distribution [11]. This means we rescale the full axion flux distribution – before taking the window strongback into account – such that it represents the fractional X-ray flux per square centimeter. That way, when we multiply it with the rest of the expression in the signal calculation eq. 12.1, the result is the expected number of counts at the given position and energy per cm².

The window strongback is not part of the simulation, because for the position uncertainty, we need to move the axion image without moving the strongback. As such the strongback is added as part of the limit calculation based on the physical position on the chip of a given candidate.

12.8.7. Computing the total signal - s_{tot}

As mentioned in sec. 12.3 in principle we need to integrate the signal function $s(E, x, y)$ over the entire chip area and all energies. However, we do not actually need to perform that integration, because we know the efficiency of our telescope and detector setup as well as the amount of flux entering the telescope.

Therefore we compute s_{tot} via

$$s_{\text{tot}}(g_{ae}^2) = \int_0^{E_{\text{max}}} f(g_{ae}^2, E) \cdot A \cdot t \cdot P_{a\mapsto\gamma}(g_{a\gamma}^2) \cdot \varepsilon(E)\ \mathrm{d}E,$$

making use of the fact that the position dependent function $r(x, y)$ integrates to 1 over the entire axion image. This allows us to precompute the integral and only rescale the result for the current coupling constant g_{ae}^2 via

[11] A big reason for this approach is that so far I have not been able to reproduce the reflectivity (and thus effective area) of the telescope to a sufficient degree. A pure raytracing approach would overestimate the amount of flux currently.

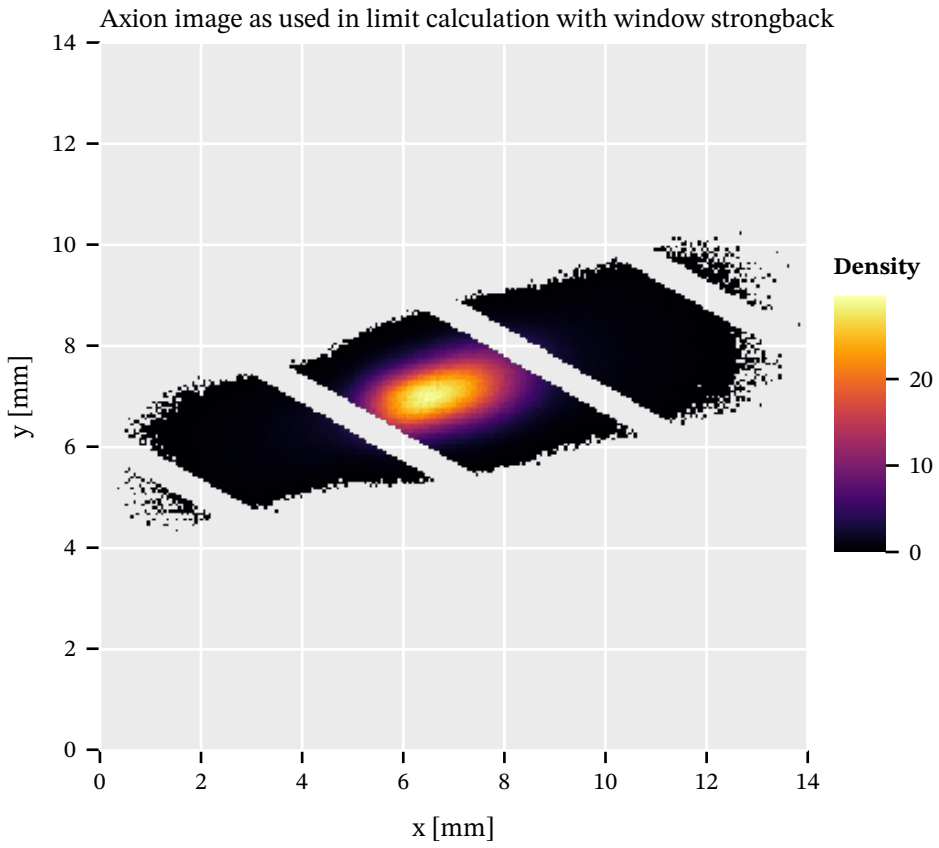


Figure 12.5.: Axion image as computed using raytracing for the AGSS09 [25, 203] solar model and under the assumption that the axion-electron coupling constant $g_{ae} = 10^{-13}$ dominates over the axion-photon coupling $g_{a\gamma} = 10^{-12}\,\text{GeV}^{-1}$. The diagonal lines with missing flux are the detector window strongbacks. It is very slightly asymmetric, because of being 0.7 cm in front of the focal point.

$$s_{\text{tot}}(g_{ae}^2) = s_{\text{tot}}(g_{ae,\text{ref}}^2) \cdot \frac{g_{ae}^2}{g_{ae,\text{ref}}^2},$$

where $g_{ae,\text{ref}}^2$ is the reference coupling constant for which the integral is computed initially. Similar rescaling needs to be done for the axion-photon coupling or chameleon coupling, when computing a limit for either.

12.8.8. Background

The background must be evaluated at the position and energy of each cluster candidate. As the background is not constant in energy or position on the chip (see sec. 11.6), we need a continuous description in those dimensions of the background rate.

In order to obtain such a thing, we start from all X-ray like clusters remaining after background rejection, see for example fig. 11.28, and construct a background interpolation. We define b_i as a function of candidate position x_i, y_i and energy E_i,

$$b_i(x_i, y_i, E_i) = \frac{I(x_i, y_i, E_i)}{W(x_i, y_i, E_i)}.$$

where I is an intensity defined over clusters within a range R and a normalization weight W. From here on we will drop the candidate suffix i. The arguments will be combined to vectors

$$\mathbf{x} = \begin{pmatrix} \vec{x} \\ E \end{pmatrix} = \begin{pmatrix} x \\ y \\ E \end{pmatrix}.$$

The intensity I is given by

$$I(\mathbf{x}) = \sum_{b \in \{\mathcal{D}(\mathbf{x}_b, \mathbf{x}) \leq R\}} \mathcal{M}(\mathbf{x}_b, \mathbf{x}) = \sum_{b \in \{\mathcal{D}(\mathbf{x}_b, \mathbf{x}) \leq R\}} \exp\left[-\frac{1}{2}\mathcal{D}^2/\sigma^2\right],$$

where we introduce $\mathcal{M}$ to refer to a normal distribution-like measure and $\mathcal{D}$ to a custom metric (for clarity without arguments). All background clusters $\mathbf{x}_b$ within some 'radius' R contribute to the intensity I, weighted by their distance to the point of interest $\mathbf{x}$. The metric is given by

$$\mathcal{D}(\mathbf{x}_1, \mathbf{x}_2) = \mathcal{D}((\vec{x}_1, E_1), (\vec{x}_2, E_2)) = \begin{cases} (\vec{x}_1 - \vec{x}_2)^2 \text{ if } |E_1 - E_2| \leq R \\ \infty \text{ if } (\vec{x}_1 - \vec{x}_2)^2 > R^2 \\ \infty \text{ if } |E_1 - E_2| > R \end{cases}$$

with $\vec{x} = \begin{pmatrix} x \\ y \end{pmatrix}$. Note first of all that this effectively describes a cylinder. Any point inside $|\vec{x}_1 - \vec{x}_2| \leq R$ simply yields a euclidean distance, as long as their energy is smaller than R. Further note, that the

distance is only dependent on the distance in the x-y plane, *not* their energy difference. Finally, this requires rescaling the energy as a common number R, but in practice the implementation of this custom metric simply compares energies directly, with the 'height' in energy of the cylinder expressed as ΔE.

The commonly used value for the radius R in the x-y plane are $R = 40\,\text{pixel}$ and in energy $\Delta E = \pm 0.3\,\text{keV}$. The standard deviation of the normal distribution for the weighting in the measure σ is set to $\frac{R}{3}$. The basic idea of the measure is simply to provide the highest weight to those clusters close to the point we evaluate and approach 0 at the edge of R to avoid discontinuities in the resulting interpolation.

Finally, the normalization weight W is required to convert the sum of I into a background rate. It is the 'volume' of our measure within the boundaries set by our metric $\mathcal{D}$:

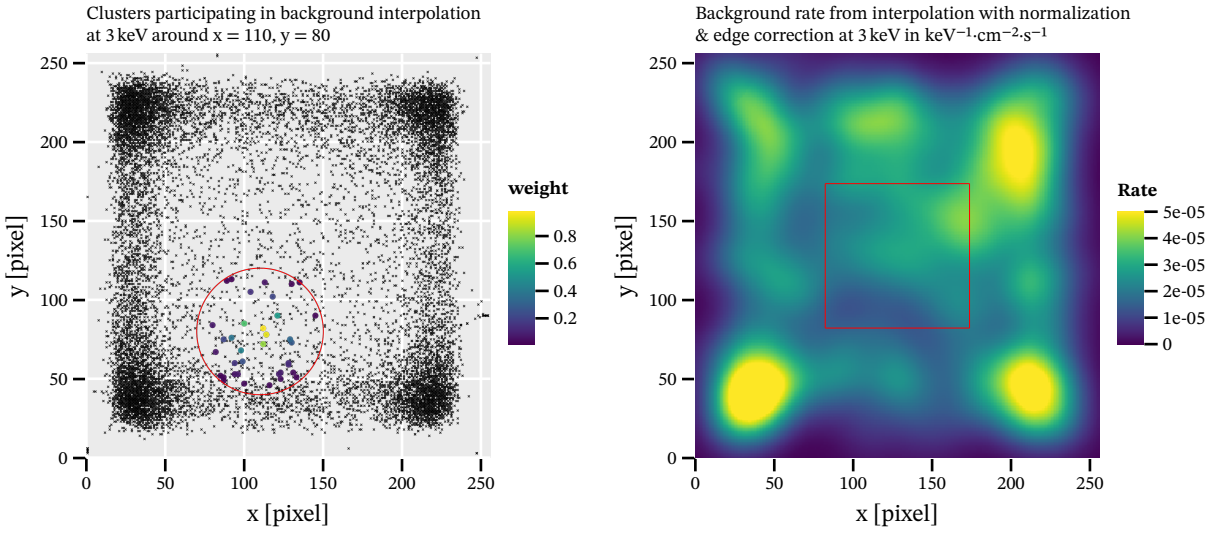
$$\begin{aligned}
W(x', y', E') &= t_B \int_{E'-E_c}^{E'+E_c} \int_{\mathcal{D}(\vec{x}', \vec{x}) \leq R} \mathcal{M}(x', y') \,\mathrm{d}x \,\mathrm{d}y \,\mathrm{d}E \\
&= t_B \int_{E'-E_c}^{E'+E_c} \int_{\mathcal{D}(\vec{x}', \vec{x}) \leq R} \exp\left[-\frac{1}{2}\mathcal{D}^2/\sigma^2\right] \mathrm{d}x \,\mathrm{d}y \,\mathrm{d}E \\
&= t_B \int_{E'-E_c}^{E'+E_c} \int_0^R \int_0^{2\pi} r \exp\left[-\frac{1}{2}\frac{\mathcal{D}^2}{\sigma^2}\right] \mathrm{d}r \,\mathrm{d}\varphi \,\mathrm{d}E \\
&= t_B \int_{E'-E_c}^{E'+E_c} -2\pi \left(\sigma^2 \exp\left[-\frac{1}{2}\frac{R^2}{\sigma^2}\right] - \sigma^2\right) \mathrm{d}E \\
&= -4\pi t_B E_c \left(\sigma^2 \exp\left[-\frac{1}{2}\frac{R^2}{\sigma^2}\right] - \sigma^2\right),
\end{aligned}$$

where we made use of the fact that within the region of interest $\mathcal{D}'$ is effectively just a radius r around the point we evaluate. t_B is the total active background data taking time. If our measure was $\mathcal{M} = 1$, meaning we would count the clusters in $\mathcal{D}(\vec{x}, \vec{x}') \leq R$, the normalization W would simply be the volume of the cylinder.

This yields a smooth and continuous interpolation of the background over the entire chip. However, towards the edges of the chip it underestimates the background rate, because once part of the cylinder is not contained within the chip, fewer clusters contribute. For that reason we correct for the chip edges by upscaling the value within the chip by the missing area. See appendix M.

Fig. 12.6 shows an example of the background interpolation centered at 3 keV, with all clusters within a radius of 40 pixels and in an energy range from 2.7 to 3.3 keV. Fig. 12.6(a) shows the initial step of the interpolation, with all colored points inside the circle being clusters that are contained in $\mathcal{D} \leq R$. Their color represents the weight based on the measure $\mathcal{M}$. After normalization and calculation for each point on the chip, we get the interpolation shown in fig. 12.6(b).

Implementation wise, as the lookup of the closest neighbors in general is an N^2 operation for N clusters, all clusters are stored in a k-d tree, for fast querying of clusters close to the point to be evaluated. Furthermore, because the likelihood $\mathcal{L}$ is evaluated many times for a given set of candidates to compute a limit, we perform caching of the background interpolation values for each candidate. That way we only compute the interpolation once for each candidate.



(a) Intensity at a point

(b) Interpolation

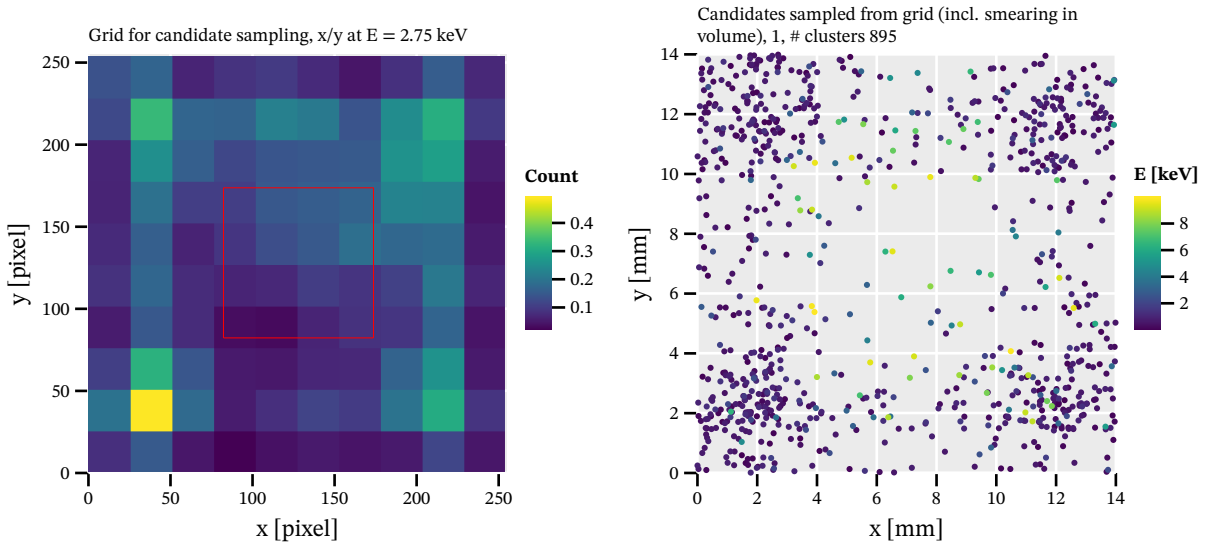
Figure 12.6.: (a): Calculation of intensity I at the center of the red circle. Black crosses indicate all background clusters. The red circle indicates cutoff R in the x-y plane. Only clusters with colored dots inside the circle are within 2.70 to 3.3 keV. Their color is the weight based on the gaussian measure $\mathcal{M}$. (b): Example of the resulting background interpolation at 3 keV computed over the entire chip. A smooth, correctly normalized interpolation is obtained.

12.8.9. Candidates

Finally, the candidates are the X-ray like clusters remaining after the background rejection algorithm has been applied to the data taken during the solar tracking. For the computation of the expected limit, the set of candidates is drawn from the background rate distribution via sampling from a Poisson distribution with the mean of the background rate. As our background model is an interpolation instead of a binned model with Poisson distributed bins, we create a grid of $(x, y, E) = (10, 10, 20)$ cells from the interpolation, which we scale such that they contain the expected number of candidates from each cell after the solar tracking duration, b_{ijk}. Then we can walk over the entire grid and sample from a Poisson distribution for each grid cell with mean $\lambda_{ijk} = b_{ijk}$. For all sampled candidates κ_{ijk} in each grid cell, we then compute a random position and energy from uniform distributions along each dimension.

A slice of the grid cells centered at 2.75 keV is shown in fig. 12.7(a), with the color indicating how many candidates are expected in each cell after the solar tracking duration.

A set of toy candidates generated in this manner is shown in fig. 12.7(b). Each point represents one toy candidate at its cluster center position. The color scale represents the energy of each cluster in keV.



(a) Grid of expected counts

(b) Toy candidate set

Figure 12.7.: (a): Expected counts in $(10, 10, 20)$ cells, centered around 2.75 keV obtained from background interpolation and normalized back to counts in solar tracking within the volume of the grid cell. (b): A set of toy candidates drawn from cells of expected counts using a Poisson distribution with mean based on each grid cell. Each point is the center of a cluster with the color scale showing the energy of that cluster.

12.9. Systematics

As explained previously in sec. 12.6, we introduce 4 different nuisance parameters to handle systematics. These are split by those impacting the signal, the background and each position axis independently.

Tab. 12.1 shows the different systematic uncertainties we consider, whether they affect signal, background or the position, their value and finally potential biases due to some imperfect knowledge. Note that the listed software efficiency systematic is an upper bound. The explicit value depends on the parameter setup for which we compute a limit, as each setup with differing software efficiency can have differing uncertainties. Further note that the accuracy given is purely the result of our estimation on the signal or background of the underlying systematic assuming some uncertainty. It does not strictly speaking reflect our knowledge of it to that extent.

All individual systematic uncertainties are combined in the form of a euclidean distance

$$\bar{\sigma} = \sqrt{\sum_i \sigma_i^2}$$

for each type of systematic (s, b). The combined uncertainties come out to

$$\begin{aligned} \sigma_s &\leq 3.38\,\% \text{ (assuming } \sigma_{\text{software}} = 2\,\%) \\ \sigma_b &= 0.28\,\% \\ \sigma_{xy} &= 5\,\% \text{ (fixed, uncertainty numbers are bounds)} \end{aligned}$$

where again the final σ_s depends on the specific setup and the given value is for a case of $\sigma_{\text{software}} = 2\,\%$, which is a bound larger than the observed uncertainties. The position uncertainty is fixed by hand to 5 % due to lack of knowledge about parameters that could be used to calculate a specific value. The numbers in the table represent bounds about the maximum deviation possible. For a derivation of these numbers, see the extended thesis [12].

Uncertainty	s or b?	rel. σ [%]	bias?
Earth ⇔ Sun distance	s	0.7732	none
Window thickness (± 10 nm)	s	0.5807	none
Solar models	s	< 1	none
Magnet length (- 1 cm)	s	0.2159	likely 9.26 m
Magnet bore diameter (± 0.5 mm)	s	2.32558	measurements: 42.x - 43
Window rotation (30° ± 0.5°)	s	0.18521	none
Software efficiency	s	< 2	none
Gas gain time binning	b	0.26918	to 0
Reference dist interp (CDL morphing)	b	0.0844	none
Alignment (signal, related mounting)	s (pos.)	0.5 mm	none
Detector mounting precision (±0.25 mm)	s (pos.)	0.25 mm	none

Table 12.1.: Overview of the different systematics that are considered as well as possible biases due to our understanding.

[12] I did not put them into the general appendix, because mostly they are small to medium large pieces of code, which simply run the relevant calculation with slightly different parameters and in the end computes the ratio of the result to the unchanged parameter result.

12.10. MCMC to sample the distribution and compute a limit

The Metropolis-Hastings algorithm [152, 103] – as mentioned in sec. 12.7 – is used to evaluate the integral over the nuisance parameters to get the posterior likelihood.

Instead of building a very long MCMC, we opt to construct 3 Markov chains with 150 000 links to reduce the bias introduced by the starting parameters. Ideally, one would construct even more chains, but a certain number of steps from the starting parameter are usually needed to get into the parameter space of large contributions to the integral (unless the starting parameters are chosen in a very confined region, which itself is problematic in terms of bias). These are removed as the 'burn in' and make the number of chains and links in each chain a trade off.

The MCMC is built based on 5 dimensional vectors $\vec{x}$,

$$\vec{x} = \begin{pmatrix} g_{ae}^2 & \vartheta_s & \vartheta_b & \vartheta_x & \vartheta_y \end{pmatrix}^T$$

containing the coupling constant of interest squared as the first entry and the four nuisance parameters after. Here we mention the axion-electron coupling constant g_{ae}^2, but generally it can also be for example $g_{ae}^2 \cdot g_{a\gamma}^2$ (equivalent to g_{ae}^2!), $g_{a\gamma}^4$ or β_γ^4, depending on the search to be conducted. The important point is that the parameter is used, under which the likelihood function is *linear*, as we otherwise bias our sampling (see the extended thesis for a longer explanation).

Our initial starting vector $\vec{x}_i$ is randomly sampled by

$$\vec{x} = \begin{pmatrix} \text{rand}([0,5]) \cdot g_{\text{base}}^2 \\ \text{rand}([-0.4, 0.4]) \\ \text{rand}([-0.4, 0.4]) \\ \text{rand}([-0.5, 0.5]) \\ \text{rand}([-0.5, 0.5]) \end{pmatrix}$$

where rand refers to a uniform random sampler in the given interval and g_{base} is a reference coupling parameter of choice, which also depends on the specific search.

Our default reference coupling constant for g_{base}^2 [13] is $g_{ae}^2 = 10^{-21}$, allowing for a range of parameters in the expected parameter space. The nuisance parameters are allowed to vary in a large region, given the standard deviations of $\sigma < 0.05$ for all four nuisance parameters. In the updating stage to propose a new vector, we use the following:

[13] Do not confuse g_{base} with the reference coupling constant for which the axion flux is computed g_{ref} mentioned earlier.

$$\vec{x_{i+1}} = \vec{x_i} + \begin{pmatrix} \text{rand}([-0.5 \cdot 3g^2_{\text{base}}, 0.5 \cdot 3g^2_{\text{base}}]) \\ \text{rand}([-0.5 \cdot 0.025, 0.5 \cdot 0.025]) \\ \text{rand}([-0.5 \cdot 0.025, 0.5 \cdot 0.025]) \\ \text{rand}([-0.5 \cdot 0.05, 0.5 \cdot 0.05]) \\ \text{rand}([-0.5 \cdot 0.05, 0.5 \cdot 0.05]) \end{pmatrix}$$

This combination leads to an acceptance rate of the new proposal typically between 20 to 30 %. After all three chains are built, the first 50 000 links each are thrown out as burn-in to make sure we only include meaningful parameter space.

The parameter space for each of the 5 elements is restricted based on the following

$$\begin{pmatrix} g = [0, \infty] \\ \vartheta_s = [-1, 1] \\ \vartheta_b = [-0.8, 1] \\ \vartheta_x = [-1, 1] \\ \vartheta_y = [-1, 1] \end{pmatrix},$$

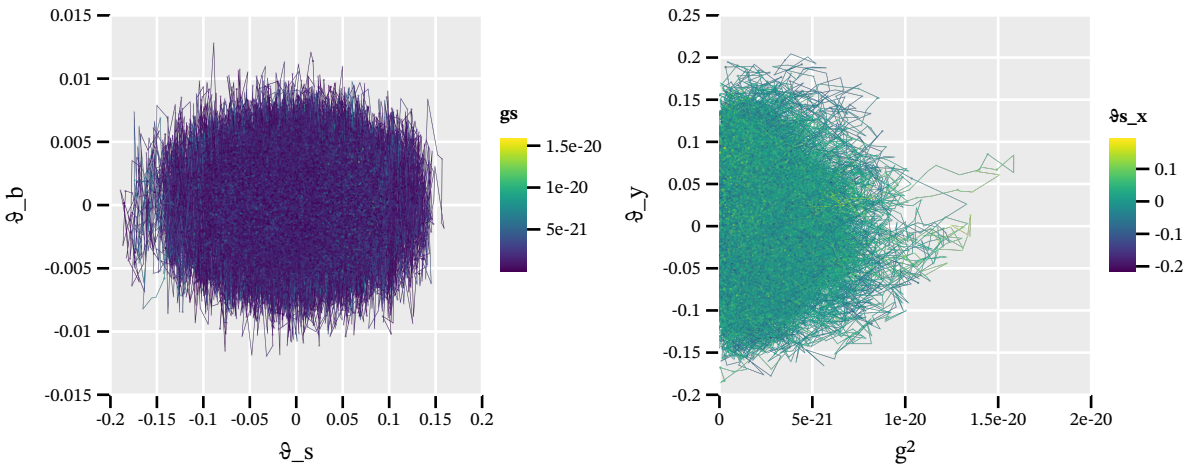
meaning we restrict ourselves to physical coupling constants and give loose bounds on the nuisance parameters. In particular for the ϑ_b parameter the restriction to values larger than $\vartheta_b > -0.8$ is due to the singularity in $\mathcal{L}_M$ at $\vartheta_b = -1$. For all realistic values for the systematic uncertainty σ_b the region of $\vartheta_b \ll 1$ has no physical meaning anyway. But for unit tests and sanity checks of the implementation, larger uncertainties are tested for, which cause computational issues if this restriction was not in place.

Example Markov chains can be seen in fig. 12.8 where we see the different nuisance parameters of the chain and how they are distributed. As expected for our comparatively small values of σ, the chain is centered around 0 for each nuisance parameter. And the coupling constant in fig. 12.8(b) also shows a clear increase towards low values.

The resulting three Markov chains are finally used to compute the marginal posterior likelihood function by computing the histogram of all sampled g^2 values. The distribution of the sampled g^2 values is that of the marginal posterior likelihood. This then allows to compute a limit by computing the empirical distribution function of the sampled g^2 values and extracting the value corresponding to the 95^{th} percentile. An example for this is shown in fig. 12.9.

12.11. Expected limits of different setups

One interesting approach to compute the expected limit usually employed in binned likelihood approaches is the so called 'Asimov dataset'. The idea is to compute the limit based on the dataset,



(a) 🐼 ϑ_s vs. ϑ_b (b) g^2 vs. ϑ_y

Figure 12.8.: (a): MCMC of the ϑ_s nuisance parameter against ϑ_b with the coupling constant as the color scale. (b): MCMC of the coupling constant against ϑ_x and ϑ_y in color. Both show a clear centering to values around 0, with the coupling constant a decrease in population towards larger couplings.

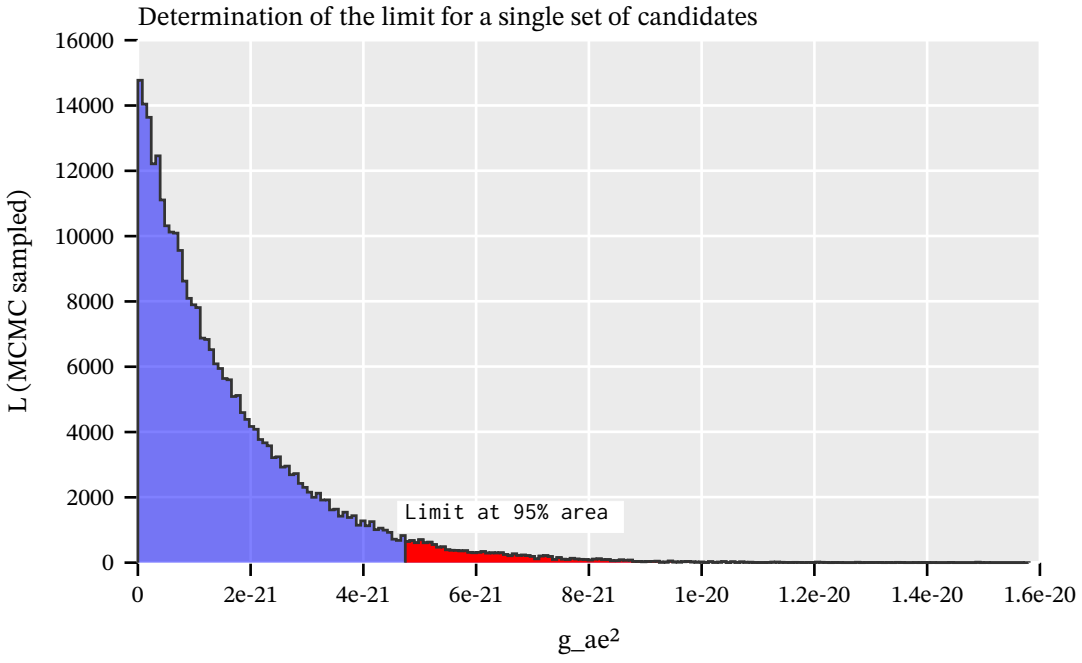


Figure 12.9.: Example likelihood as a function of g_{ae}^2 for a set of toy candidates with the limit indicated at the intersection of the blue and red colored areas. Blue is the lower 95-th percentile of the integral over the likelihood function and red the upper 5-th.

which matches exactly the expectation value for each Poisson in each bin. This has the tremendous computational advantage of providing an expected limit by only computing the limit for a *single, very special* candidate set. Unfortunately, for an unbinned approach this is less straightforward, because there is no explicit mean of expectation values anymore. On the other hand, the Asimov dataset calculation does not provide information about the possible spread of all limits due to the statistical variation possible in toy candidate sets.

In our case then, we fall back to computing an expected limit based on toy candidate sets (sec. 12.5) that we draw from a discretized, grid version of the background interpolation, as explained in sec. 12.8.9.

We compute expected limits for different possible combinations of classifier and detector vetoes, due to the signal efficiency penalties that these imply. In a first rough 'scan' we compute expected limits based on a smaller number of toy candidates. The exact number depends on the particular setup.

- 1000 toys for setups using only a classifier without any vetoes. Due to the much more background towards the corners (see sec. 11.6), many more candidates are sampled making computation significantly slower
- 2500 toys for setups with the line or septem veto. Much fewer total expected number of candidates, and hence much faster.

Then we narrow it down and compute 15 000 toy limits for the best few setups. Finally, we compute 50 000 toys for the best setup we find. The resulting setup is the one for which we unblind the solar tracking data.

Tab. 12.2 shows the different setups with their respective expected limits. Some setups appear multiple times for the different number of toy candidates that were run. It can be seen as an extension of tab. 11.8 in sec. 11.6. Also shown is the limit achieved in case no candidate is present in the signal sensitive region, as a theoretical lower bound on the limit. This 'no candidate' limit scales down with the total efficiency, as one would expect. All limits are given as limits on $g_{ae} \cdot g_{a\gamma}$ based on a fixed $g_{a\gamma} = 10^{-12}\,\text{GeV}^{-1}$. Finally, it shows a standard variation corresponding to how the expected limit varies when bootstrapping new sets of limits (standard deviation of 1000 bootstrapped expected limits sampling 10 000 limits from the input). The table does not show all setups that were initially considered. Further considerations of other possible parameters were excluded in preliminary studies on their effect on the expected limit.

In particular:

- the scintillator veto is always used. It does not come with an efficiency penalty and therefore there is no reason not to activate it.

- different FADC veto efficiencies as well as disabling it completely were considered. The current $\varepsilon_{\text{FADC}} = 0.98$ efficiency was deemed optimal. Harder cuts do not yield significant improvements.
- the potential eccentricity cutoff for the line veto, as discussed in sec. 11.5.4 is fully disabled, as the efficiency gains do not outweigh the positive effect on the expected limit in practice.

Based on this study, the MLP produces the best expected limit, surprisingly without any vetoes at software efficiencies of 98.04 %, 90.59 % and 95.23 %. However, for these case we did not run any more toy limits, because not having any of the septem or line means there are a large number of candidates towards the chip corners. These slow down the calculation, making it too costly to run. In any case, given the small difference in the expected limit between this case and the first including vetoes, the MLP at 95.23 % with the line veto, we prefer to stick with the addition of vetoes. The very inhomogeneous background rates are problematic, as they make the result much more strongly dependent on the value of the systematic position uncertainty. Also, for other limit calculations with larger raytracing images, a lower background over a larger area at the cost of lower total efficiency is more valuable.

In this case with the line veto, the software efficiency ε_{eff} corresponds to a target software efficiency based on the simulated X-ray data of 95 %. The total combined efficiency comes out to 79.69 % (excluding the detection efficiency of course!). This is the setup we will mainly consider for the data unblinding. The expected limit for this setup is

$$\left(g_{ae} \cdot g_{a\gamma}\right)_{\text{expected}} = 7.8782(65) \times 10^{-23}\,\text{GeV}^{-1},$$

based on the 50 000 toy limits. The uncertainty is based on the standard deviation computed via bootstrapping as mentioned above. It is the uncertainty on the expected limit from a statistical point of view.

ε_{eff}	nmc	Type	Septem	Line	$\varepsilon_{\text{total}}$	No signal [GeV^{-1}]	Expected [GeV^{-1}]	Exp. σ [GeV^{-1}]
0.9804	1000	MLP	false	false	0.9804	5.739e-23	7.805e-23	3.6807e-25
0.9059	1000	MLP	false	false	0.9059	6.0109e-23	7.856e-23	4.301e-25
0.9523	1000	MLP	false	false	0.9523	5.7685e-23	7.8599e-23	5.1078e-25
0.9523	2500	MLP	false	true	0.7969	6.3874e-23	7.8615e-23	2.9482e-25
0.9523	50000	MLP	false	true	0.7969	6.3874e-23	7.8782e-23	6.4635e-26
0.9804	2500	MLP	false	true	0.8204	6.1992e-23	7.8833e-23	2.9977e-25
0.8587	1000	MLP	false	false	0.8587	6.1067e-23	7.9597e-23	5.0781e-25
0.9059	2500	MLP	false	true	0.7581	6.4704e-23	7.9886e-23	2.6437e-25
0.9804	2500	MLP	true	true	0.7554	6.5492e-23	8.0852e-23	2.9225e-25
0.9523	2500	MLP	true	false	0.7756	6.4906e-23	8.1135e-23	3.5689e-25

Continued on next page

Continued from previous page

ε_{eff}	nmc	Type	Septem	Line	$\varepsilon_{\text{total}}$	No signal [GeV^{-1}]	Expected [GeV^{-1}]	Exp. σ [GeV^{-1}]
0.9523	2500	MLP	true	true	0.7338	6.6833e-23	8.1251e-23	3.0965e-25
0.9804	2500	MLP	true	false	0.7985	6.2664e-23	8.1314e-23	3.1934e-25
0.8587	2500	MLP	false	true	0.7186	6.8094e-23	8.1561e-23	2.9893e-25
0.9059	2500	MLP	true	false	0.7378	6.5184e-23	8.2169e-23	2.8767e-25
0.9	2500	LnL	false	true	0.7531	6.4097e-23	8.2171e-23	3.7248e-25
0.9059	2500	MLP	true	true	0.6981	6.8486e-23	8.2868e-23	3.2593e-25
0.8587	2500	MLP	true	false	0.6994	6.7322e-23	8.4007e-23	2.9498e-25
0.9	2500	LnL	true	true	0.6935	6.7386e-23	8.4274e-23	3.3644e-25
0.8587	2500	MLP	true	true	0.6617	6.9981e-23	8.4589e-23	3.4966e-25
0.8	2500	LnL	false	true	0.6695	6.9115e-23	8.4993e-23	3.1983e-25
0.9	2500	LnL	false	false	0.9	5.9862e-23	8.5786e-23	3.7241e-25
0.8	2500	LnL	false	false	0.8	6.3885e-23	8.7385e-23	3.903e-25
0.8	2500	LnL	true	true	0.6165	7.1705e-23	8.747e-23	4.099e-25
0.7	2500	LnL	false	true	0.5858	7.4553e-23	8.9298e-23	4.0495e-25
0.7	2500	LnL	false	false	0.7	6.7647e-23	9.0856e-23	3.3235e-25
0.7	2500	LnL	true	true	0.5394	7.7018e-23	9.2565e-23	3.4573e-25

Table 12.2.: The FADC veto is always in use (if either the septem or line veto are active) at an efficiency of $\varepsilon_{\text{FADC}} = 0.98$ and so is the scintillator veto. These settings were defined in preliminary studies of the expected limits. Note the efficiencies associated with the septem veto $\varepsilon_{\text{septem}} = 83.11\,\%$ and the line veto $\varepsilon_{\text{line}} = 85.39\,\%$ and combined $\varepsilon_{\text{septem+line}} = 78.63\,\%$, which are implicitly included based on the 'Septem' and 'Line' column values into the total efficiency $\varepsilon_{\text{total}}$. 'No signal' is the limit without any candidates, 'Exp. σ' the bootstrapped standard deviation of the expected limit.

The distribution of all toy limits for this best setup can be seen in fig. 12.10. It shows both the limit for the case without any candidates (red line, equivalent to 'No signal' in the table above) as well as the expected limit (blue line). Depending on the number of candidates that are inside the signal sensitive region (in regions of the solar axion image with significant flux expectation) based on $\ln(1 + s_i/b_i) > 0.5$ (at a fixed coupling constant, $g_{ae}^2 = (8.1 \times 10^{-11})^2$), the limits are split into histograms of different colors. Based on the location of these histograms and the expected limit, the most likely case for the real candidates seems to be 1 or 2 candidates in that region. Note that there are some toy limits that are below the red line for the case without candidates. This is expected, because the calculation of each limit is based on the MCMC evaluation of the likelihood. As such it is a statistical random process and the red line itself is a single sample. Further, the purple histogram for "0" candidates is **not** equivalent to the red line, because the definition of the number of signal sensitive candidates is an arbitrary cutoff. For the red line literally no candidates at all are considered and the limit is based purely on the $\exp(-s_{\text{tot}})$ term of the likelihood.

To see how the limit might change as a function of different candidates, see tab. N.1 in appendix

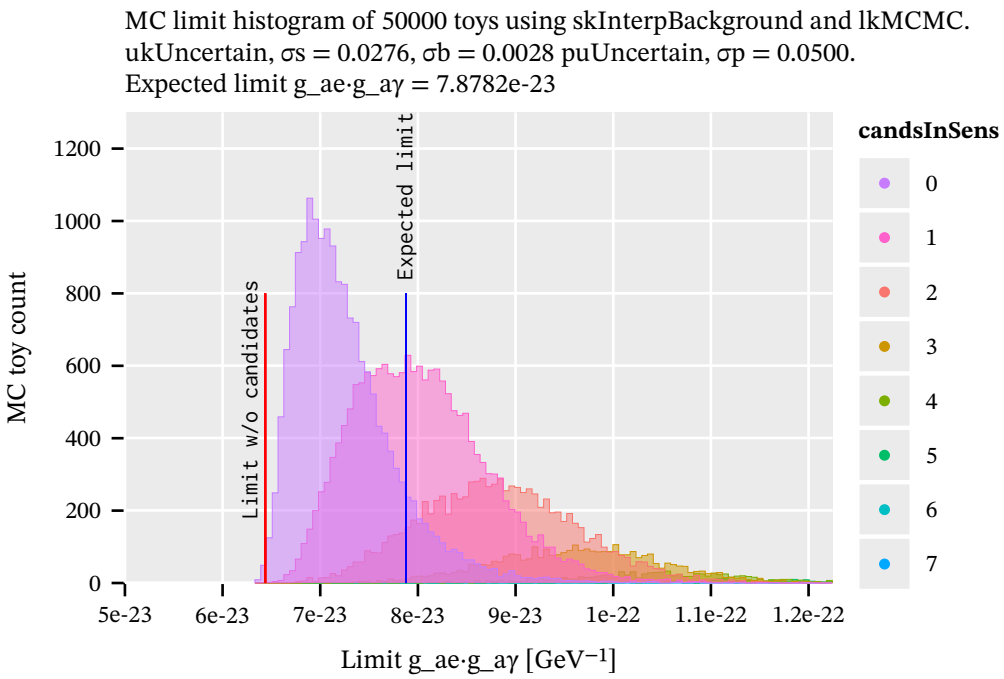


Figure 12.10.: Distribution of toy limits resulting in the best expected limit based on 50 000 toy limits. The expected limit – the median – is shown as the blue line. The red line shows the limit for the case without any candidates. The different colored histograms correspond to toy sets with a different number of toy candidates in the signal sensitive region, defined by $\ln(1 + s_i/b_i) > 0.5$. The most likely number of candidates in the sensitive region seems to be 0, 1 or 2.

N.2. It contains different percentiles of the computed toy limit distribution for each veto setup. The percentiles – and ranges between them – give insight into the probabilities to obtain a specific observed limit. Each observed limit is associated with a single possible set of candidates, measureable in the experiment, out of all possible sets of candidates compatible with the background hypothesis (as the toy limits are sampled from it). For example, the experiment will measure an observed limit in the range from P_{25} to P_{75} with a chance of 50 %.

12.11.1. Verification

Because of the significant complexity of the limit calculation, a large number of sanity checks were written. They are used to verify all internal results are consistent with expectation. They include things like verifying the background interpolation reproduces a compatible background rate or the individual s_i terms of the likelihood reproduce the total s_{tot} term, while producing sensible numbers.

The details of this verification are left out of the main thesis, but can be found in the extended version after this section.

12.12. Solar tracking candidates

Based on the best performing setup we can now look at the solar tracking candidates. [14] In this setup, based on the background model a total of 845 candidates are expected over the entire chip. [15] Computing the real candidates yields a total of 850 clusters over the chip. A figure comparing the rate observed for the candidates compared to the background over the entire chip is shown in fig. 12.12(b).

Fig. 12.11 shows all solar tracking candidates that are observed with the method yielding the best expected limit. Their energy is color coded and written above each cluster within a 85 pixel radius of the chip center. The axion image is underlaid to provide a visual reference of the importance of each cluster. Very few candidates of relevant energies are seen within the region of interest. Based on the previously mentioned $\ln(1 + s_i/b_i) > 0.5$ condition, it is 1 candidate in the sensitive region. See fig. 12.12(a) for an overview of the weighting of each candidate in this way, with only the single cluster near coordinate $(x, y) = (105, 125)$ crossing the threshold of 0.5.

[14] The actual data unblinding of the candidates presented in this section was only done after the analysis of the previous sections was fully complete. A presentation with discussion took place first inside our own group and later with the relevant members of the CAST collaboration to ascertain that our analysis appears sound.

[15] The background model contains 16 630 clusters in this case. 3156.8 h of background data and 160.375 h of tracking data yields $16\,630 \cdot \frac{160.375}{3156.8} \approx 845$ clusters.

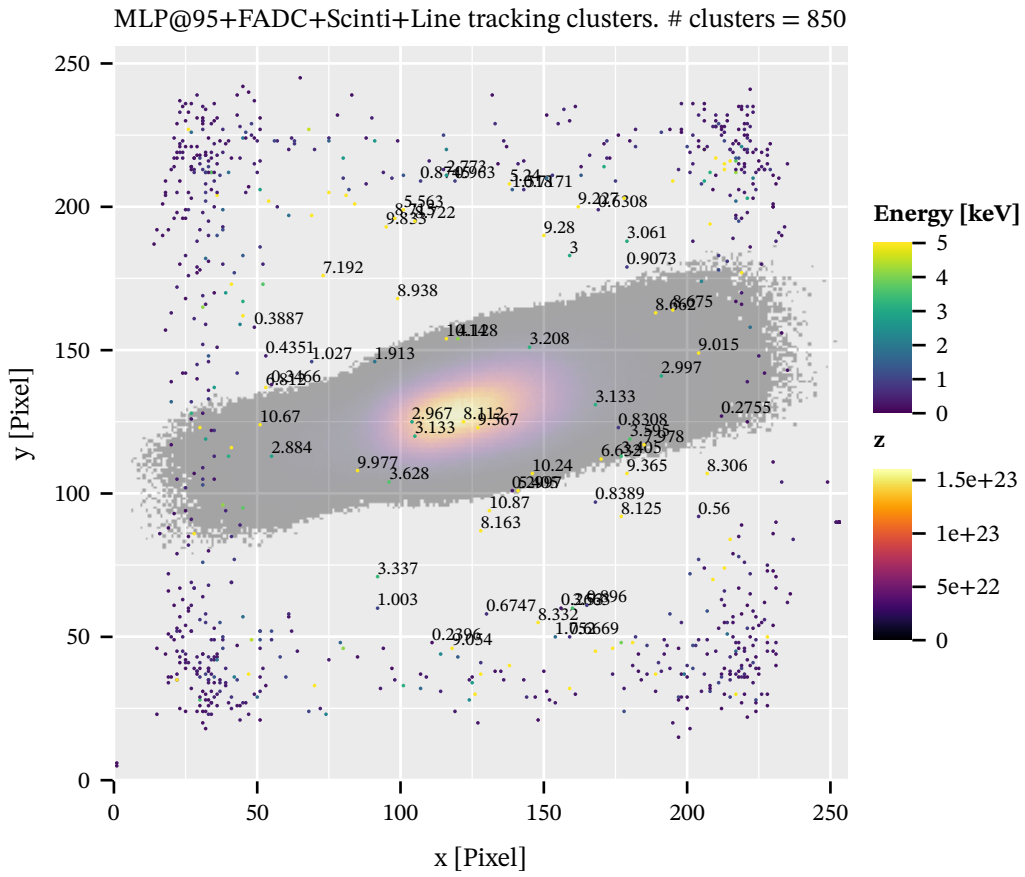
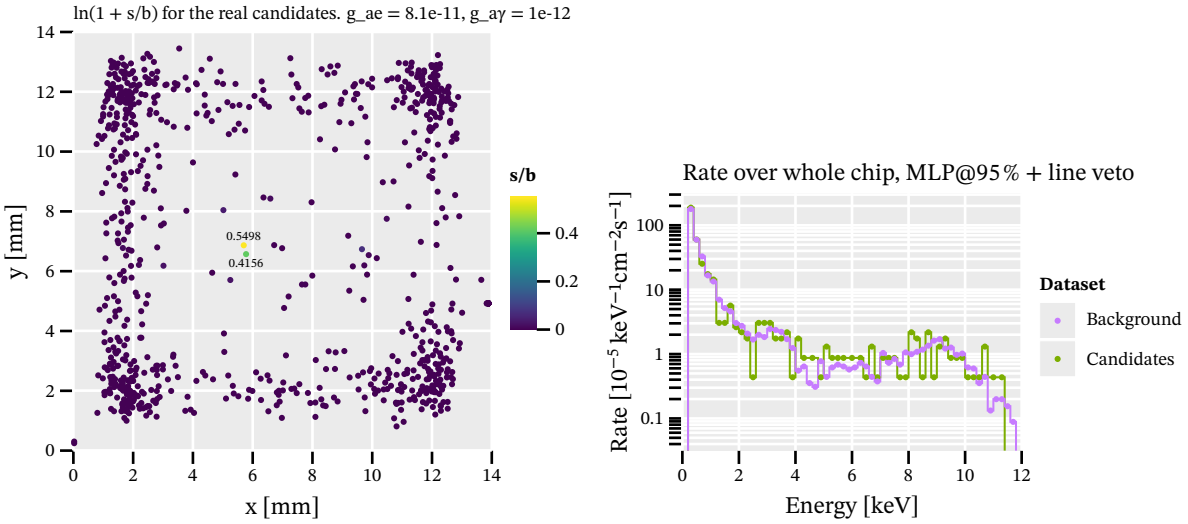


Figure 12.11.: Overview of all solar tracking candidates given the MLP@95 % setup including all vetoes except the septem veto. 850 clusters are observed, with the majority low energy clusters in the chip corners. The axion image is underlaid and the energy of each cluster is color coded. For all candidates within a radius of 85 pixels of the center the energy is also written above. We can see by eye that very few candidates of relevant energies are present in the region of expected signal.



(a) $\ln(1 + s_i/b_i)$ (b) Rate candidates vs. background

Figure 12.12.: (a): Weighting of each candidate based on $\ln(1+s_i/b_i)$. The largest weight given is for the 2.96 keV cluster seen in fig. 12.11 near pixel (105, 125) and tops out at slighlty above 0.5. (b): Rate of background and candidate clusters over the entire chip as a log plot. Rates are generally compatible over the entire range.

12.13. Observed limit - g_{ae}

With the data fully unblinded and the solar tracking candidates known, we can now compute the observed limit for the axion-electron coupling g_{ae}. We compute an observed limit of

$$\left(g_{ae} \cdot g_{a\gamma}\right)_{\text{observed}} = 7.34(9) \times 10^{-23}\,\text{GeV}^{-1},$$

which is lower than the expected limit, due to the distribution of candidates and absence of any highly significant candidates. This limit is the mean value out of 200 limits computed via 3 Markov Chains of 150 000 links (same as for the expected limits) computed for the real candidates. The printed uncertainty represents the standard deviation out of those limits. Therefore, we may wish to present an upper bound,

$$\left(g_{ae} \cdot g_{a\gamma}\right)_{\text{observed}} \lesssim 7.35 \times 10^{-23}\,\text{GeV}^{-1} \text{ at } 95\,\%\text{ CL}.$$

The expected limit for this case was $\left(g_{ae} \cdot g_{a\gamma}\right)_{\text{expected}} = 7.8782(65) \times 10^{-23}\,\text{GeV}^{-1}$ and the limit without any candidates at all $\left(g_{ae} \cdot g_{a\gamma}\right)_{\text{no candidates}} = 6.39 \times 10^{-23}\,\text{GeV}^{-1}$.

This is a good improvement compared to the current, best observed limit by CAST in 2013 [37], which achieved

$$\left(g_{ae} \cdot g_{a\gamma}\right)_{\text{CAST2013}} \lesssim 8.1 \times 10^{-23}\,\text{GeV}^{-1}.$$

Unfortunately, [37] does not provide an expected limit to compare to. [16]

Fig. 12.13 shows the marginal posterior likelihood function for the observed solar tracking candidates, for a single calculation run (out of the 200 mentioned). The limit is at the 95$^{\text{th}}$ percentile of the histogram, shown by the intersection of the blue and red filling. In addition the yellow line shows values based on a numerical integration using Romberg's method [182] at 20 different coupling constants. This is a cross validation of the MCMC result. [17]

Note that this observed limit is valid for axion masses in the range where the coherence condition in the conversion probability is met. That is, $qL \ll \pi$, refer back to equation 3.2. This holds up to axion masses around $m_a \lesssim 0.02\,\text{eV}$, but the exact value is both energy dependent and based on the desired cutoff in reduction of the conversion probability. See fig. N.1 in appendix N.1 to see how the conversion probability develops as a function of axion mass. The expected and observed limits simply (inversely) follow the conversion probability, i.e. out of coherence they get exponentially worse, superimposed with the periodic modulation seen in the conversion probability. As we did not perform a buffer gas run, the behavior in that range is not computed, because it is mostly trivial (only the exact point at which the limit decreases changes depending on the exact energies of the candidates).

Finally, note that if one combines the existing astrophysical limits on g_{ae} alone (for example tip of the red giant branch star brightness limits, [57] at $g_{ae} < 1.3 \times 10^{-13}$) with an axion-photon coupling of choice (for example the current best limit of [64]) one may very well obtain a 'better' limit on $g_{ae} \cdot g_{a\gamma}$. In that sense the above, at the very least, represents the best helioscope limit on the product of both coupling constants. It also suffers less from uncertainties as astrophysical limits, see for example [76].

12.14. Other coupling constants

As explained when we introduced the limit calculation method, one important aim was to develop a method which is agnostic to the coupling constant of choice. We will now make use of this to compute expected and observed limits for the axion-photon coupling $g_{a\gamma}$, sec. 12.14.1 and chameleon coupling β_γ, sec. 12.14.2.

[16] Judging by the χ^2 distribution in [37], fig. 6, having a minimum for negative $g_{ae}^2 g_{g\gamma}^2$ values, potentially implies a better observed limit than expected.

[17] Note though, while the calculation of the observed limit via the MCMC takes about 10 s, the numerical integration using Romberg's method takes $\sim$ 1 h for only 20 points. And that is only using an integration level of 5 (a parameter of the Romberg method, one often uses 8 for Romberg for accuracy). This highlights the need for Monte Carlo methods, especially for expected limits.

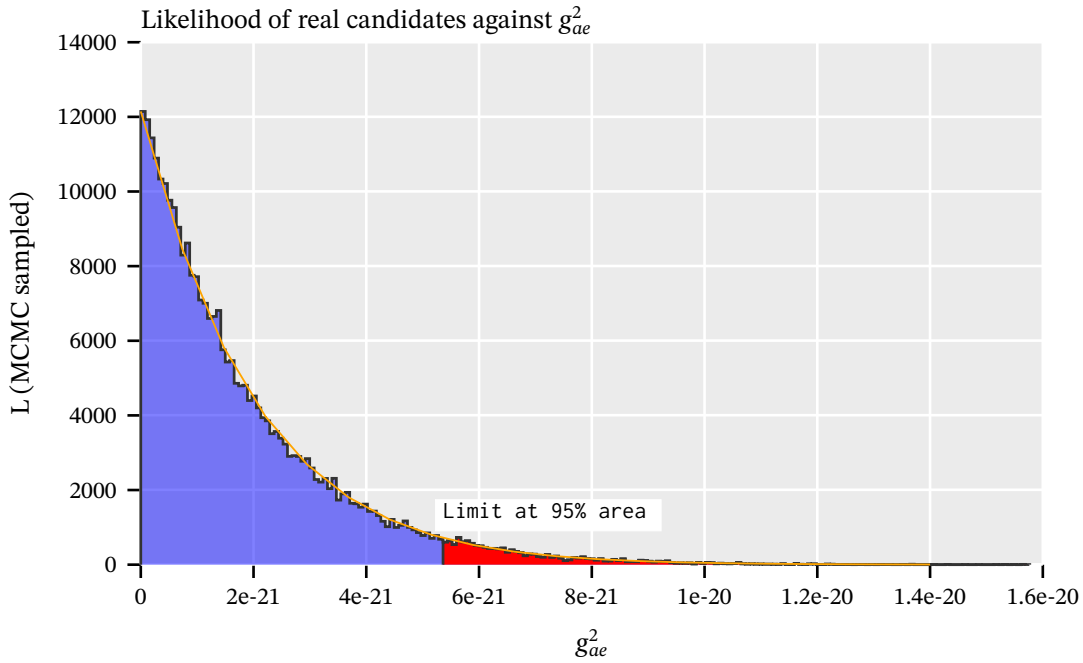


Figure 12.13.: The marginal posterior likelihood function for the solar candidates with the observed limit at the intersection between the blue and red filling. Also shown is a line based on a numerical integration of the 4-fold integral at 20 steps as a cross check of the MCMC. The x-axis is the parameter we sample, namely g_{ae}^2. Limit at $g^2 \approx 5.4 \times 10^{-21} \Rightarrow g \approx 7.35 \times 10^{-11}$.

12.14.1. Axion-photon coupling - $g_{a\gamma}^4$

The axion-photon coupling requires the following changes:

- use $g_{a\gamma}^4$ in the vector of the Markov Chain, replacing the g_{ae}^2 term. As $g_{a\gamma}$ both affects the production and reconversion it needs to be in the fourth power.
- the axion flux based on Primakoff production only.
- the axion image based on the Primakoff flux only.

The axion flux and axion image for the Primakoff production are shown in fig. 12.14. Note that the axion image is computed at the same effective conversion position as for the axion-electron flux. Strictly speaking this is not quite correct, due to the different energies of the two fluxes and therefore different absorption lengths. For the purposes here the inaccuracy is acceptable.

Based on these we compute an expected limit for the same setup that yielded the best expected limit for the axion-electron coupling constant. Namely, the MLP classifier at 95 % efficiency using all vetoes except the septem veto. We compute an expected limit based on 10^4 toy candidates. Fig. 12.15 shows the distribution of limits obtained for different toy candidates, including the expected

limit. In appendix N.2, tab. N.2 the different percentiles for this distribution are shown. The obtained expected limit is

$$g_{a\gamma,\text{expected}} = 9.0650(75) \times 10^{-11}\,\text{GeV}^{-1},$$

which compared to the observed CAST Nature [64] limit of $g_{a\gamma,\text{Nature}} = 6.6 \times 10^{-11}\,\text{GeV}^{-1}$ is of course significantly worse. This is expected however, due to significantly less tracking time and higher background rates. The limit without any candidates comes out to

$$g_{a\gamma,\text{no candidates}} = 7.95 \times 10^{-11}\,\text{GeV}^{-1}.$$

Based on the same candidates as in sec. 12.13 we obtain an observed limit of

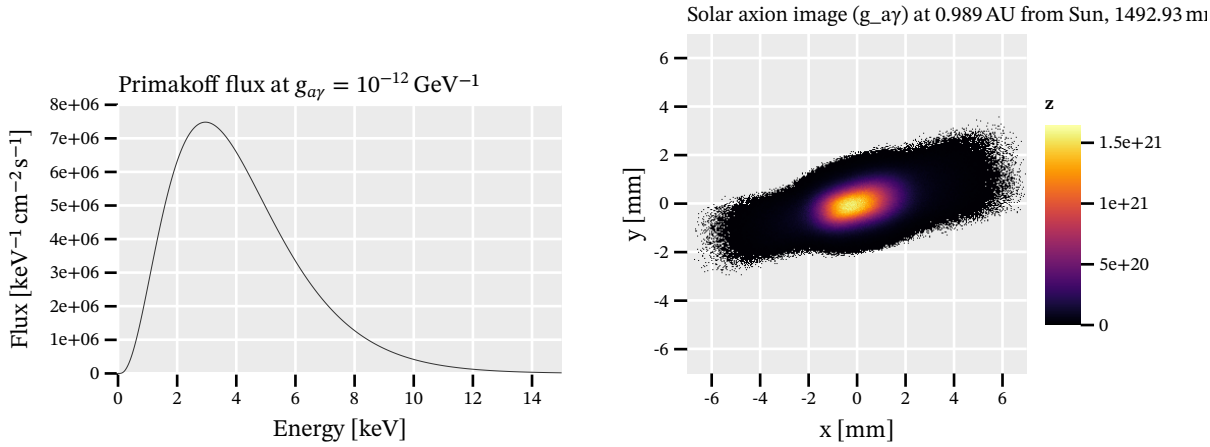
$$g_{a\gamma,\text{observed}} = 8.99(7) \times 10^{-11}\,\text{GeV}^{-1},$$

which again is the mean out of 200 MCMC limits. Once again it is better than the expected limit, similar as for the axion-electron limit, but the overall result is as expected.

As a bound then, it is

$$g_{a\gamma,\text{observed}} \lesssim 9.0 \times 10^{-11}\,\text{GeV}^{-1} \text{ at } 95\,\%\text{ CL}.$$

The distribution of the posterior likelihood function can be found in fig. N.2 of appendix N.3.



(a) $g_{a\gamma}$ flux

(b) $g_{a\gamma}$ image

Figure 12.14.: (a): Differential axion flux arriving on Earth due to Primakoff production assuming $10^{-12}\,\text{GeV}^{-1}$. (b): Axion image for the Primakoff emission in the Sun.

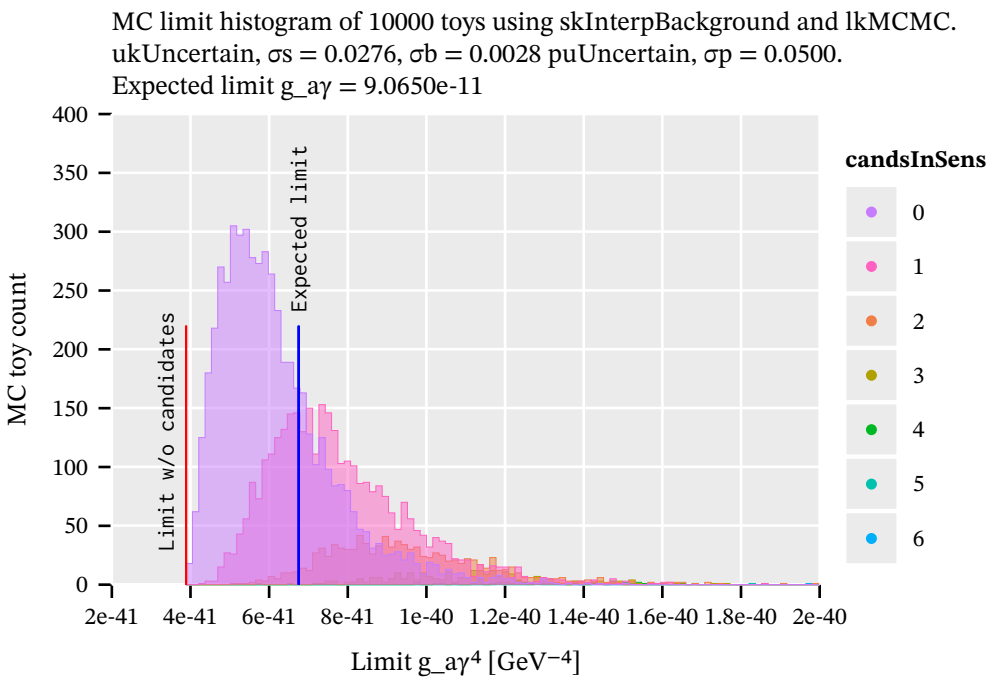


Figure 12.15.: 10^4 toy limits for the axion-photon coupling $g_{a\gamma}^4$. The expected limit is determined to $g_{a\gamma,\text{expected}} = 9.0650(75) \times 10^{-11}\,\text{GeV}^{-1}$, with the no candidates limit being $g_{a\gamma,\text{no candidates}} = 7.95 \times 10^{-11}\,\text{GeV}^{-1}$.

12.14.2. Chameleon coupling - β_γ^4

Let's now have a look at the chameleon coupling. In addition to the inputs and MCMC parameter that needs to be changed similarly to the axion-photon coupling (flux, image and using β_γ^4 in the MCMC), the conversion probability needs to be adjusted according to eq. 3.8. This assumes the conversion is fully coherent and we restrict ourselves to non-resonant production. That means a chameleon-matter coupling β_m of

$$1 \leq \beta_m \leq 10^6.$$

Further, because the chameleon production occurs in the solar tachocline – at around $0.7 \cdot R_\odot$ – the angles under which chameleons can reach the CAST magnet are much larger than for axions. This leads to a significant fraction of chameleons not traversing the entire magnet. For any chameleon traversing only parts of it, its probability for conversion is decreased. According to [9] this is accounted for by a correction factor of 38.9 % in reduction of signal. [134] used a simplified raytracing model for this. While the distance through the magnet can easily be modeled with our raytracer (see appendix R), we will still use the correction factor [18].

[18] Including the conversion probability from the raytracer would ideally mean to include the reflectivities under angles encountered for each ray. However, attempts to reproduce the effective area as provided by LLNL raytracing simulations have failed so far. In order to avoid complications with potential inaccuracies, we stick to the previous approach for simplicity and better comparison.

Fig. 12.16 shows the differential chameleon flux, assuming a magnetic field of 10 T at the solar tachocline region and using $\beta_\gamma = \beta_\gamma^{\text{sun}} = 6.46 \times 10^{10}$ (the bound on the chameleon coupling from solar physics). The chameleon image is in stark contrast to the very small axion image seen in the previous sections. The outer most ring corresponds to production in the tachocline regions in which our view is effectively tangent to the tachocline normal (i.e. the 'outer ring' of the solar tachocline when looking at the Sun). Also visible is the asymmetry in the signal on the chip, due to the LLNL telescope. The densest flux regions are in the top and bottom. These correspond to the narrow sides of the ellipsoid for example in fig. 12.14(b). This 'focusing effect' was not visible in the raytracing simulation of [134], due to the simpler raytracing approach, which approximated the ABRIXAS telescope as a lens (slight differences between ABRIXAS and LLNL telescope would of course exist).

The relative size of the chameleon image compared to the size of the GridPix was one of the main motivators to implement the background interpolation as introduced in sec. 12.8.8. This allows us to utilize the entire chameleon flux and weigh each chameleon candidate correctly. This is a significant improvement compared to the 2014/15 detector result [134], in which the entire outer ring of the chameleon flux had to be dropped. Not only can we include these regions in our calculation due to our approach, but also our background level is *significantly* lower in these outer regions thanks to the usage of our vetoes (compare fig. 11.29, where 11.29(a) is comparable to the background level of [134]).

Again, we follow the approach used for the axion-photon coupling and restrict ourselves to a single calculation of an expected limit for the best veto setup (MLP at 95 % efficiency using vetoes except the septem veto) based on 10^4 toy candidate sets. This yields an expected limit of

$$\beta_{\gamma,\text{expected}} = 3.6060(39) \times 10^{10}.$$

Without any candidates, which for the chameleon due to its large much larger focused image is significantly more extreme, the limit comes out to

$$\beta_{\gamma,\text{no candidates}} = 2.62 \times 10^{10}.$$

Fig. 12.17 shows the histograms of these toy limits. The differently colored histograms are again based on an arbitrary cutoff in $\ln(1 + s/b)$ for a fixed coupling constant. We can see that the difference between the 'no candidate' limit and the lowest toy limits is much larger than for the two axion limits. This is due to the chameleon image covering a large fraction of the chip, making it incredibly unlikely to have no candidates. Further, in appendix N.2 we find tab. N.3, containing the different percentiles for the distribution of toy limits.

For the chameleon coupling it may be worthwhile to investigate other veto setups again, because of the very different nature of the chameleon image and even lower peak of the differential flux. Based

on the same setup we compute an observed limit using the same set of candidates as previously of

$$\beta_{\gamma,\text{observed}} = 3.10(2) \times 10^{10}.$$

or as an upper bound

$$\beta_{\gamma,\text{observed}} \lesssim 3.1 \times 10^{10} \text{ at } 95\,\% \text{ CL}.$$

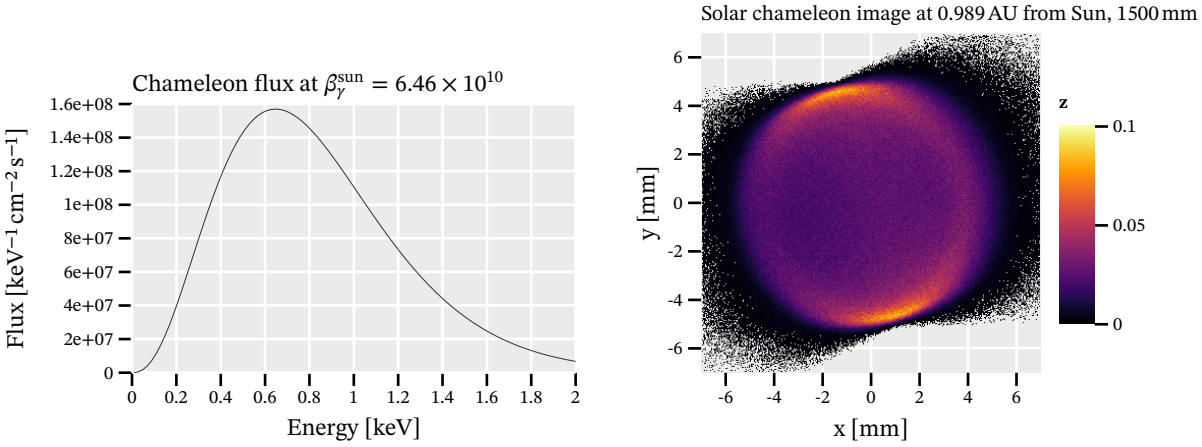
This is a good improvement over the limit of [134, 8], and the current best chameleon-photon bound, of

$$\beta_{\gamma,\text{Krieger}} = 5.74 \times 10^{10} \text{ at } 95\,\% \text{ CL}.$$

despite significantly less solar tracking time than in the former (our 160 h to 254 h in [134]), thanks to the significant improvements in background due to the detector vetoes, higher detection efficiency (thinner window), better classifier and improved limit calculation method allowing for the inclusion of the entire chameleon flux.

A figure of the sampled coupling constants, similar to fig. 12.13 can be found in appendix N.4, fig. N.3.

Note however, that it is somewhat surprising that the observed limit is also an improvement over the expected limit, as the total number of clusters in the tracking data is almost exactly as expected (850 for 844 expected). Based on all evaluations I have done, it seems to be a real effect of the candidates. In fig. 12.12(b) of the spectrum comparing candidates to background, we see that there is a slight, but noticeable lower rate at energies below 2 keV, which is the relevant range for chameleons. As such it may very well be a real effect, despite putting it below the 5^{th} percentile of the toy limits (compare tab. N.3). Further investigation seems appropriate.



(a) β_γ flux (b) β_γ image

Figure 12.16.: (a): Differential chameleon flux arriving on Earth assuming $\beta_\gamma^{\text{sun}} = 6.46 \times 10^{10}$ and a magnetic field of $B = 10\,\text{T}$ in the solar tachocline. (b): Chameleon image for chameleon emission in the Sun.

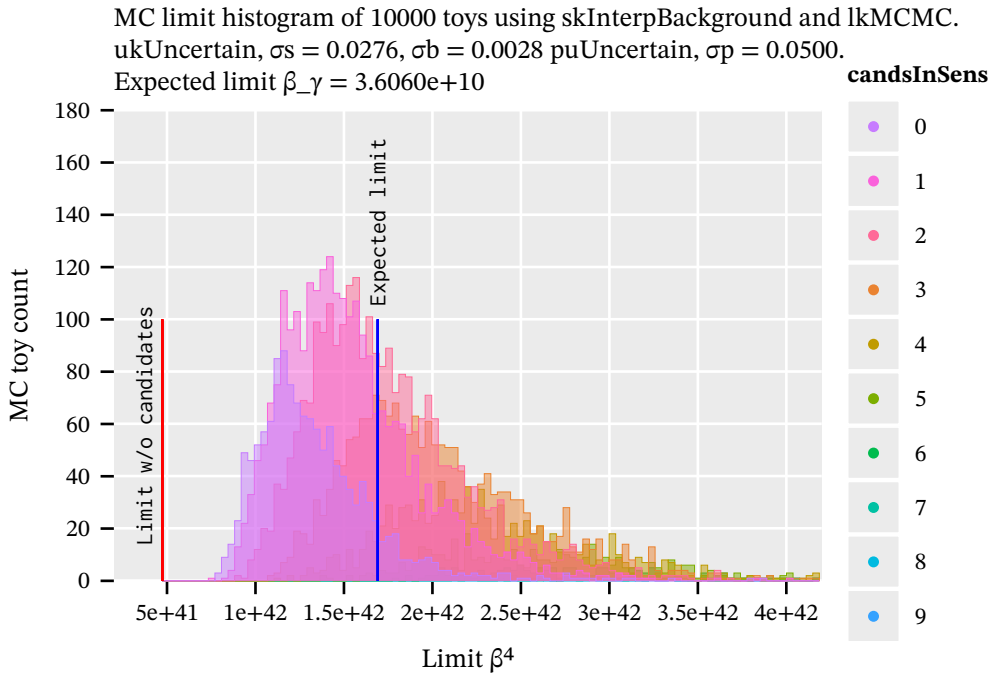


Figure 12.17.: 10^4 toy limits for the chameleon coupling β_γ^4. The expected limit is determined to $\beta_{\gamma,\text{expected}} = 3.6060(39) \times 10^{10}$, with the no candidates limit being $\beta_{\gamma,\text{no candidates}} = 2.62 \times 10^{10}$.

13. Outlook

Having computed a new best limit on the axion-electron and chameleon couplings, as well as presented a limit on the axion-photon coupling, let us think about the next steps that should be taken from here.

The first obvious step would be computing a combined limit of the dataset from this thesis with the GridPix data from 2014/15 [134]. As the old detector used the same readout system, the entire analysis framework used in this thesis is directly compatible with the old data. This would be straightforward and should lead to a decent improvement on the limit, in particular due to the good expected limits for the MLP classifier at high software efficiencies, as there are no vetoes available for the old detector.

Secondly, a combined limit of all GridPix data with all Micromegas data should be computed. At least at the likelihood level – meaning to multiply the posterior likelihood functions for each detector and dataset – would be little work and could lead to another limit improvement. While this would imply exclusion of systematic uncertainties for non GridPix data, in theory one could even attempt to compute limits with the MCMC code written for this thesis. GridPix and Micromegas detector differ, but fundamentally at the level of the limit calculation both work with cluster center positions and energies. This would require the corresponding systematic uncertainties and detection efficiencies.

Focusing on a new detector, there are two major improvements on the horizon. First of all a future detector is likely going to be built from radiopure materials. This should reduce the inherent background seen in the detector significantly. In addition, such a detector will be built on top of the Timepix3 instead of the Timepix. This comes with massive improvements in terms of achievable background rates at higher signal efficiencies. The Timepix3 can be read out in a stream-based fashion. Therefore, there is no more dead time associated with a readout and long shutter times are a thing of the past. More importantly, this means for a 7-GridPix detector layout, the very high random coincidence rates seen for the current detector (see sec. 11.5.5) are completely removed. For a Timepix3 based detector the septem veto and line veto can be used *without any efficiency penalty*. Combined with an MLP classifier as used in this thesis and without considering radiopure materials, background rates in the 5×10^{-6} to $7 \times 10^{-6}\,\mathrm{keV^{-1}\,cm^{-2}\,s^{-1}}$ range are achievable at software efficiencies around the 95 % mark (see tab. 11.8 and tab. 12.2). This alone would lead to a significant improvement in limit calculations. Furthermore, the Timepix3 supports reading out the Time

over Threshold values *in addition to* the Time of Arrival (ToA) values. At that point the FADC is not needed anymore to provide time information nor to act as a readout trigger. All noise related issues due to dealing with analogue signals in such an experiment are side-stepped. Further, the availability of pixel-wise ToA values allows to reconstruct events in three dimensions, making the equivalent of the rise time veto for the FADC much more accurate. Indeed, in a prototype Timepix3 based detector utilizing an upper cut on the ToA values of clusters already shows promising results [188].

One positive aspect of a new Timepix3 based detector is that the full analysis procedure explained in the context of this thesis is already written with Timepix3 based detectors in mind. While some additions will likely be desired, the basis is already there and given a dataset for a Timepix3 based detector a limit calculation could be done in a matter of days. [1]

In terms of a possible setup of such a Timepix3 based detector at BabyIAXO in the future, a few things are important. Firstly, a much improved scintillator veto setup will greatly reduce the possible background due to muons or X-ray fluorescence. One thing to consider for this though, is to move away from a purely discriminator based system. The scintillators used in this thesis were much less useful than initially hoped, due to lack of interpretability of the data and imperfect calibrations. Ideally, the analogue signals of each scintillator are read out so that discriminator thresholds can be applied offline during data analysis (while this introduces analogue signals back into the system, the highly amplified nature makes this significantly easier to deal with). This makes the system much more flexible and less error prone in terms of choosing the wrong parameters for a data taking campaign.

Next, given that a Timepix3 based detector will surely need a cooling system as well, time should be invested in characterizing its performance and possibly even working towards a temperature stabilized system. Generally, more sensors should be installed, both for temperature measurements in different places, as well as pressures to make correlating detector behaviors with external parameters easier.

On the data calibration side, more importance should be placed on measurements behind an X-ray tube. Data quality in the CAST Detector Lab (CDL) data for this detector was unfortunately not ideal and generally more statistics should be available. While probably unreasonable, BabyIAXO should have a calibration source installed on the opposite side of the magnet, which can produce different X-ray energies. This would allow calibrations both through the X-ray telescope for more frequent alignment measurements as well as X-ray calibration data taken at the same conditions as the background and tracking data. The discrepancy between CAST data detector behavior and CDL data is larger than I would like.

Moreover, simulations of a future detector should be higher priority in the future. Both in terms

[1] This is also one additional reason why I felt it important to provide a thesis that is fully reproducible. This makes it easier for other people to actually use the developed software and understand the methods.

of general background studies and electric field simulations, as well as event simulations. Having more realistic synthetic datasets would be invaluable for better developments of classifiers *without* the need to rely on real datasets. This was already very successful with the comparatively simple synthetic X-rays in this thesis, but could be extended for background data in the future for pure synthetically trained machine learning classifiers. In this vein more powerful approaches would become feasible, especially if more calibration data were available as well. Convolutional Neural Networks are a promising candidate to investigate, especially also for the 3D cluster data of a Timepix3 detector.

And hey, maybe soon we can also compute a limit on the Axio-Chameleon [50]?

14. Summary & conclusion

In this thesis we have presented the data taking campaign of the 7-GridPix 'Septemboard' detector used at CAST in 2017/18, its data analysis and limit calculations for different coupling constants.

In total about 3150 h of active background and 160 h of active solar tracking data were taken at CAST with this detector. Generally, detector performance was very stable, with a few issues likely relating to varying operating conditions. There were some other minor setbacks during CAST data taking, most notably a ruptured window, which delayed data taking by a short amount. Also the FADC setup was partially affected by significant noise, which was fixed by changing amplifier settings. In addition, the scintillator data was not recorded correctly in the Run-2 data taking period (Oct 2017 to Apr 2018) and temperature log files of the detector were mostly lost. In the grand scheme of things these issues are minor and do not affect the physics potential of the data much.

We have shown that the additional detector features are an extremely valuable addition. Most notably the Septemboard itself, in the form of the 'septem veto' and 'line veto', provide a large improvement to the backgrounds seen over the majority of the center GridPix. While the improvements to the center $5 \times 5\,\mathrm{mm}^2$ region are not as large, over the entire chip background is suppressed by an order of magnitude.

In the course of the thesis, the data reconstruction and analysis code written `TimepixAnalysis` [194], was written with future Timepix3 based detectors in mind. It is ready for usage for such detectors.

Further, novel ideas were implemented to improve the reliability in the form of interpolating the reference datasets the likelihood cut classifier is based on. More interestingly though, a machine learning approach using a small multi-layer perceptron (MLP) was implemented, trained entirely on synthetic X-ray data and background data of the outer chips. This yields a classifier fully defined by data unrelated to its main application (background and tracking datasets) and verification (^{55}Fe calibration and X-ray tube data). This MLP classifier achieves comparable performance to the likelihood cut at its default $\varepsilon_{\ln\mathcal{L}} = 80\,\%$ using a software efficiency of 95 %. Significant improvements to the limit calculation are possible as a result.

Following the 2017 CAST Nature paper [64], an unbinned Bayesian likelihood method for limit calculation was implemented to compute limits on the axion-electron coupling g_{ae}, the axion-photon coupling $g_{a\gamma}$ and the chameleon coupling β. This limit calculation requires a description of the irre-

ducible background during data taking and several inputs to calculate the expected number of axion induced X-rays during the solar tracking time. The background is obtained from the application of the classifier ($\ln \mathcal{L}$ or MLP) to the background dataset.

To calculate the expected number of axion induced X-rays during the solar tracking dataset, the differential solar axion flux and the radial emission profile in the Sun is needed (in addition to the losses expected due to the detector window losses and gas transmission). To properly characterize the expected 'axion image' on the detector, a raytracing simulation taking these into account is required. Such a simulation was implemented in the context of this thesis and verified against PANTER measurements of the LLNL telescope.

Despite the additional detector features producing a more homogeneous background rate over the entire central chip, an interpolation of the background rate at each point is still required. Such an interpolation was developed based on a normal distribution weighted nearest neighbor approach, producing smooth results.

Further, the limit calculation method of [64] was extended to allow the inclusion of systematic uncertainties into the likelihood function by usage of four nuisance parameters. As these need to be integrated out to obtain the posterior likelihood function from which a limit is computed, a Markov Chain Monte Carlo (MCMC) approach was developed to sample from the likelihood function efficiently. This is needed, because different choices of parameters (detector vetoes, software efficiencies and so on) are evaluated based on their resulting expected limit. Expected limits are computed by sampling toy candidate sets from the background distribution and computing their limits. The expected limit then is defined by the median of all such limits.

The expected limit for the best method based on 50 000 toy candidate sets came out to

$$\left(g_{ae} \cdot g_{a\gamma}\right)_{\text{expected}} = 7.8782(65) \times 10^{-23}\,\text{GeV}^{-1},$$

while the observed limit was computed to

$$\left(g_{ae} \cdot g_{a\gamma}\right)_{\text{observed}} \lesssim 7.35 \times 10^{-23}\,\text{GeV}^{-1} \text{ at } 95\,\%\text{ CL}.$$

This is a good improvement over the current best limit obtained by CAST in 2013 [37] of

$$\left(g_{ae} \cdot g_{a\gamma}\right)_{\text{CAST2013}} \lesssim 8.1 \times 10^{-23}\,\text{GeV}^{-1}.$$

For the axion-photon coupling our detector was not expected to improve on the current best limit ($g_{a\gamma,\text{Nature}} < 6.6 \times 10^{-11}\,\text{GeV}^{-1}$, [64]), which is validated by an expected limit of

$$g_{a\gamma,\text{expected}} = 9.0650(75) \times 10^{-11}\,\text{GeV}^{-1},$$

with an observed limit of

$$g_{a\gamma,\text{observed}} \lesssim 9.0 \times 10^{-11}\,\text{GeV}^{-1} \text{ at } 95\,\%\text{ CL}.$$

However, for the chameleon limit this detector was expected to be highly competitive, due to a 300 nm thin Si_3N_4 window greatly increasing efficiency at the required low energies and additional detector vetoes improving background rates over the entire center GridPix. Indeed, these combined with an improved limit calculation method, allowing the inclusion of the entire center chip and thus full chameleon flux, and better classifier in the form of the MLP yields an expected limit of

$$\beta_{\gamma,\text{expected}} = 3.6060(39) \times 10^{10}.$$

The observed limit was then computed to

$$\beta_{\gamma,\text{observed}} \lesssim 3.1 \times 10^{10} \text{ at } 95\,\%\text{ CL}.$$

which is a significant improvement over [134, 8]

$$\beta_{\gamma,\text{Krieger}} < 5.74 \times 10^{10} \text{ at } 95\,\%\text{ CL},$$

the current best bound on the chameleon-photon coupling.

Bibliography

[1] G. 't Hooft. "Symmetry Breaking through Bell-Jackiw Anomalies." In: *Phys. Rev. Lett.* 37 (1 July 1976), pp. 8–11. DOI: 10.1103/PhysRevLett.37.8. URL: http://link.aps.org/doi/10.1103/PhysRevLett.37.8.

[2] J. Aalbers et al. "A Search for New Physics in Low-Energy Electron Recoils From the First Lz Exposure." In: (2023). DOI: 10.48550/ARXIV.2307.15753. URL: https://arxiv.org/abs/2307.15753.

[3] C. M. Adair et al. "Search for Dark Matter Axions With Cast-Capp." In: *Nature Communications* 13.1 (2022), p. 6180. DOI: 10.1038/s41467-022-33913-6. URL: http://dx.doi.org/10.1038/s41467-022-33913-6.

[4] Peter Shirley Adam Marrs and Ingo Wald, eds. *Ray Tracing Gems II*. http://raytracinggems.com/rtg2. Apress, 2021. DOI: 10.1007/978-1-4842-7185-8.

[5] S. Agostinelli et al. "GEANT4–a simulation toolkit." In: *Nucl. Instrum. Meth. A* 506 (2003), pp. 250–303. DOI: 10.1016/S0168-9002(03)01368-8. URL: https://geant4.cern.web.ch.

[6] G.D. Alkhazov. "Statistics of Electron Avalanches and Ultimate Resolution of Proportional Counters." In: *Nuclear Instruments and Methods* 89 (1970), pp. 155–165. DOI: 10.1016/0029-554x(70)90818-9. URL: http://dx.doi.org/10.1016/0029-554X(70)90818-9.

[7] Shunichi Amari. "A Theory of Adaptive Pattern Classifiers." In: *IEEE Transactions on Electronic Computers* EC-16.3 (1967), pp. 299–307. DOI: 10.1109/pgec.1967.264666. URL: http://dx.doi.org/10.1109/PGEC.1967.264666.

[8] V. Anastassopoulos et al. "Improved Search for Solar Chameleons With a Gridpix Detector At Cast." In: *Journal of Cosmology and Astroparticle Physics* 2019.01 (2019), pp. 032–032. DOI: 10.1088/1475-7516/2019/01/032. URL: http://dx.doi.org/10.1088/1475-7516/2019/01/032.

[9] V. Anastassopoulos et al. "Search for Chameleons With Cast." In: *Physics Letters B* 749 (2015), pp. 172–180. DOI: 10.1016/j.physletb.2015.07.049. URL: http://dx.doi.org/10.1016/j.physletb.2015.07.049.

[10] T. W. Anderson. “On the Distribution of the Two-Sample Cramer-Von Mises Criterion.” In: *The Annals of Mathematical Statistics* 33.3 (1962), pp. 1148–1159. DOI: 10.1214/aoms/1177704477. URL: http://dx.doi.org/10.1214/aoms/1177704477.

[11] S Andriamonje et al. “An improved limit on the axion–photon coupling from the CAST experiment.” In: *Journal of Cosmology and Astroparticle Physics* 2007.04 (Apr. 2007), pp. 010–010. DOI: 10.1088/1475-7516/2007/04/010. URL: https://doi.org/10.1088/1475-7516/2007/04/010.

[12] S Andriamonje et al. “Development and performance of Microbulk Micromegas detectors.” In: *Journal of Instrumentation* 5.02 (Feb. 2010), P02001–P02001. DOI: 10.1088/1748-0221/5/02/p02001. URL: https://doi.org/10.1088/1748-0221/5/02/p02001.

[13] Alexandre Beynel Antje Behrens. *CAST Alignment of InGRID detector on V2*. July 2018. URL: https://edms.cern.ch/document/2005895/1 (visited on 11/17/2022).

[14] Alexandre Beynel Antje Behrens. *CAST Measurement of MPE telescope on V1, LLNL telescope and InGRID on V2*. Apr. 2018. URL: https://edms.cern.ch/document/2006138/1 (visited on 12/18/2022).

[15] Benoit Cumer Antje Behrens. *CAST Livermore Telescope Fiducialisation on V2*. July 2017. URL: https://edms.cern.ch/document/1827959/1 (visited on 11/17/2022).

[16] Benoit Cumer Antje Behrens Alexandre Beynel. *CAST Alignment of InGRID detector on V2*. Sept. 2017. URL: https://edms.cern.ch/document/2005606/1 (visited on 11/17/2022).

[17] Benoit Cumer Antje Behrens Alexandre Beynel. *CAST Alignment of InGRID detector on V2*. Oct. 2017. URL: https://edms.cern.ch/document/2005690/1 (visited on 11/17/2022).

[18] Takahiko Aoyama. “Generalized Gas Gain Formula for Proportional Counters.” In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 234.1 (1985), pp. 125–131. DOI: 10.1016/0168-9002(85)90817-4. URL: http://dx.doi.org/10.1016/0168-9002(85)90817-4.

[19] E. Aprile et al. “Excess Electronic Recoil Events in Xenon1t.” In: *Physical Review D* 102.7 (2020), p. 072004. DOI: 10.1103/physrevd.102.072004. URL: http://dx.doi.org/10.1103/PhysRevD.102.072004.

[20] E. Aprile et al. “Search for New Physics in Electronic Recoil Data From Xenonnt.” In: *Physical Review Letters* 129.16 (2022), p. 161805. DOI: 10.1103/physrevlett.129.161805. URL: http://dx.doi.org/10.1103/PhysRevLett.129.161805.

[21] E. Aprile et al. “The Xenon1t Dark Matter Experiment.” In: *The European Physical Journal C* 77.12 (2017), p. 881. DOI: 10.1140/epjc/s10052-017-5326-3. URL: http://dx.doi.org/10.1140/epjc/s10052-017-5326-3.

[22] E Arik et al. “Probing eV-scale axions with CAST.” In: *Journal of Cosmology and Astroparticle Physics* 2009.02 (Feb. 2009), pp. 008–008. DOI: 10.1088/1475-7516/2009/02/008. URL: https://doi.org/10.1088/1475-7516/2009/02/008.

[23] M. Arik et al. “New solar axion search using the CERN Axion Solar Telescope with ^{4}He filling.” In: *Phys. Rev. D* 92 (2 July 2015), p. 021101. DOI: 10.1103/PhysRevD.92.021101. URL: https://link.aps.org/doi/10.1103/PhysRevD.92.021101.

[24] E Armengaud et al. “Conceptual design of the International Axion Observatory (IAXO).” In: *Journal of Instrumentation* 9.05 (May 2014), T05002–T05002. DOI: 10.1088/1748-0221/9/05/t05002. URL: https://doi.org/10.1088/1748-0221/9/05/t05002.

[25] Martin Asplund et al. “The Chemical Composition of the Sun.” In: *Annual Review of Astronomy and Astrophysics* 47.1 (2009), pp. 481–522. DOI: 10.1146/annurev.astro.46.060407.145222. URL: http://dx.doi.org/10.1146/annurev.astro.46.060407.145222.

[26] Peter Athron et al. “Global Fits of Axion-Like Particles To Xenon1t and Astrophysical Data.” In: *Journal of High Energy Physics* 2021.5 (2021), p. 159. DOI: 10.1007/jhep05(2021)159. URL: http://dx.doi.org/10.1007/JHEP05(2021)159.

[27] B. Al Atoum et al. “Electron Transport in Gaseous Detectors With a Python-Based Monte Carlo Simulation Code.” In: *Computer Physics Communications* 254 (2020), p. 107357. DOI: 10.1016/j.cpc.2020.107357. URL: http://dx.doi.org/10.1016/j.cpc.2020.107357.

[28] F. Aznar et al. “A Micromegas-Based Low-Background X-Ray Detector Coupled To a Slumped-Glass Telescope for Axion Research.” In: *Journal of Cosmology and Astroparticle Physics* 2015.12 (2015), pp. 008–008. DOI: 10.1088/1475-7516/2015/12/008. URL: http://dx.doi.org/10.1088/1475-7516/2015/12/008.

[29] Nigel R Badnell et al. “Updated opacities from the Opacity Project.” In: *Monthly Notices of the Royal Astronomical Society* 360.2 (2005), pp. 458–464.

[30] R Bähre et al. “Any Light Particle Search II - Technical Design Report.” In: *Journal of Instrumentation* 8.09 (2013), T09001–T09001. DOI: 10.1088/1748-0221/8/09/t09001. URL: http://dx.doi.org/10.1088/1748-0221/8/09/T09001.

[31] Justin Sillvan Baier. “Hunting chameleons with the KWISP detector at the CAST experiment at CERN.” PhD thesis. Dissertation, Universität Freiburg, 2022.

[32] C. A. Baker et al. “Improved Experimental Limit on the Electric Dipole Moment of the Neutron.” In: *Phys. Rev. Lett.* 97 (13 Sept. 2006), p. 131801. DOI: 10.1103/PhysRevLett.97.131801. URL: http://link.aps.org/doi/10.1103/PhysRevLett.97.131801.

[33] Varouzhan Baluni. “CP-nonconserving effects in quantum chromodynamics.” In: *Phys. Rev. D* 19 (7 Apr. 1979), pp. 2227–2230. DOI: 10.1103/PhysRevD.19.2227. URL: http://link.aps.org/doi/10.1103/PhysRevD.19.2227.

[34] Marco Barbera et al. “Monitoring the stability of thin and medium back-up filters of the Newton-XMM EPIC camera.” In: *X-Ray and Gamma-Ray Telescopes and Instruments for Astronomy*. Vol. 4851. SPIE. 2003, pp. 264–269. URL: https://doi.org/10.1117/12.461592.

[35] Marco Barbera et al. “The thin and medium filters of the EPIC camera on-board XMM-Newton: measured performance after more than 15 years of operation.” In: *Experimental Astronomy* 42.2 (2016), pp. 179–197. DOI: 10.1007/s10686-016-9505-2. URL: http://dx.doi.org/10.1007/s10686-016-9505-2.

[36] Roger J Barlow. *Statistics: a guide to the use of statistical methods in the physical sciences*. Vol. 29. John Wiley & Sons, 1993.

[37] K Barth et al. “CAST constraints on the axion-electron coupling.” In: *Journal of Cosmology and Astroparticle Physics* 2013.05 (May 2013), pp. 010–010. DOI: 10.1088/1475-7516/2013/05/010. URL: https://doi.org/10.1088/1475-7516/2013/05/010.

[38] August Beer. “Bestimmung der Absorption des rothen Lichts in farbigen Flüssigkeiten.” In: *Ann. Physik* 162 (1852), pp. 78–88. URL: https://books.google.com/books?id=PNmXAAAAIAAJ&pg=PA78.

[39] Howard C Berg. *Random walks in biology*. Princeton University Press, 1993.

[40] M.M. Miller Bertolami et al. “Revisiting the Axion Bounds From the Galactic White Dwarf Luminosity Function.” In: *Journal of Cosmology and Astroparticle Physics* 2014.10 (2014), pp. 069–069. DOI: 10.1088/1475-7516/2014/10/069. URL: http://dx.doi.org/10.1088/1475-7516/2014/10/069.

[41] Stephen Biagi. “Degrad, transport of electrons in gas mixtures.” In: (1995). URL: https://degrad.web.cern.ch/degrad/.

[42] Stephen Biagi. “MAGBOLTZ, transport of electrons in gas mixtures.” In: (1995). URL: https://magboltz.web.cern.ch/magboltz/.

[43] K. van Bibber et al. “Design for a practical laboratory detector for solar axions.” In: *Phys. Rev. D* 39 (8 Apr. 1989), pp. 2089–2099. DOI: 10.1103/PhysRevD.39.2089. URL: https://link.aps.org/doi/10.1103/PhysRevD.39.2089.

[44] Hans Bichsel. “A method to improve tracking and particle identification in TPCs and silicon detectors.” In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 562.1 (2006), pp. 154–197. ISSN: 0168-9002. DOI: https://doi.org/10.1016/j.nima.2006.03.009. URL: https://www.sciencedirect.com/science/article/pii/S0168900206005353.

[45] Hans Bichsel. “Straggling in Thin Silicon Detectors.” In: *Reviews of Modern Physics* 60.3 (1988), pp. 663–699. DOI: 10.1103/revmodphys.60.663. URL: http://dx.doi.org/10.1103/RevModPhys.60.663.

[46] F. Bloch. “Bremsvermögen Von Atomen Mit Mehreren Elektronen.” In: *Zeitschrift für Physik* 81.5-6 (1933), pp. 363–376. DOI: 10.1007/bf01344553. URL: http://dx.doi.org/10.1007/bf01344553.

[47] M Bona et al. *Performance of the first CERN-INFN 10 m long superconducting dipole prototype for the LHC*. Tech. rep. 1994.

[48] M. Bona et al. “Design, Fabrication Variants and Results of Lhc Twin-Aperture Models.” In: *IEEE Transactions on Magnetics* 28.1 (1992), pp. 338–341. DOI: 10.1109/20.119880. URL: http://dx.doi.org/10.1109/20.119880.

[49] Pierre Bouguer. *Essai d’optique, sur la gradation de la lumiere*. Claude Jombert, 1729. URL: https://archive.org/details/UFIE003101_TO0324_PNI-2703_000000.

[50] Philippe Brax, C. P. Burgess, and F. Quevedo. *Axio-Chameleons: A Novel String-Friendly Multi-field Screening Mechanism*. 2023. arXiv: 2310.02092 [hep-th].

[51] Philippe Brax and Anne-Christine Davis. “Distinguishing Modified Gravity Models.” In: *Journal of Cosmology and Astroparticle Physics* 2015.10 (2015), pp. 042–042. DOI: 10.1088/1475-7516/2015/10/042. URL: http://dx.doi.org/10.1088/1475-7516/2015/10/042.

[52] Philippe Brax, Axel Lindner, and Konstantin Zioutas. “Detection Prospects for Solar and Terrestrial Chameleons.” In: *Physical Review D* 85.4 (2012), p. 043014. DOI: 10.1103/physrevd.85.043014. URL: http://dx.doi.org/10.1103/PhysRevD.85.043014.

[53] Nicolai Brejnholt. “NuSTAR calibration facility and multilayer reference database: Optic response model comparison to NuSTAR on-ground calibration data: Optic response model comparison to NuSTAR on-ground calibration data.” English. PhD thesis. 2012.

[54] David A Brown et al. “ENDF/B-VIII. 0: The 8th major release of the nuclear reaction data library with CIELO-project cross sections, new standards and thermal scattering data.” In: *Nuclear Data Sheets* 148 (2018), pp. 1–142.

[55] CAEN. “FADC Technical Information Manual.” In: (2010).

[56] M. Campbell et al. “Detection of Single Electrons By Means of a Micromegas-Covered Medipix2 Pixel Cmos Readout Circuit.” In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 540.2-3 (2005), pp. 295–304. DOI: 10.1016/j.nima.2004.11.036. URL: http://dx.doi.org/10.1016/j.nima.2004.11.036.

[57] Francesco Capozzi and Georg Raffelt. “Axion and Neutrino Bounds Improved With New Calibrations of the Tip of the Red-Giant Branch Using Geometric Distance Determinations.” In: *Physical Review D* 102.8 (2020), p. 083007. DOI: 10.1103/physrevd.102.083007. URL: http://dx.doi.org/10.1103/PhysRevD.102.083007.

[58] Emile Carbone et al. “Data Needs for Modeling Low-Temperature Non-Equilibrium Plasmas: the Lxcat Project, History, Perspectives and a Tutorial.” In: *Atoms* 9.1 (2021), p. 16. DOI: 10.3390/atoms9010016. URL: http://dx.doi.org/10.3390/atoms9010016.

[59] M. Chefdeville et al. “An electron-multiplying ‘Micromegas’ grid made in silicon wafer post-processing technology.” In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 556.2 (2006), pp. 490–494. ISSN: 0168-9002. DOI: https://doi.org/10.1016/j.nima.2005.11.065. URL: https://www.sciencedirect.com/science/article/pii/S0168900205021418.

[60] Xinye Chen and Stefan Güttel. *Fast and explainable clustering based on sorting*. arXiv EPrint arXiv:2202.01456. UK: The University of Manchester, 2022, p. 25. URL: https://arxiv.org/abs/2202.01456.

[61] Per H. Christensen and Wojciech Jarosz. “The Path to Path-Traced Movies.” In: *Foundations and Trends in Computer Graphics and Vision* 10.2 (Oct. 2016), pp. 103–175. ISSN: 1572-2740. DOI: 10/gfjwjc.

[62] M. Civitani et al. “Cold and Hot Slumped Glass Optics with interfacing ribs for high angular resolution x-ray telescopes.” In: *SPIE Proceedings*. Aug. 2016. DOI: 10.1117/12.2232591. URL: http://dx.doi.org/10.1117/12.2232591.

[63] Andrew G. Cohen, David B. Kaplan, and Ann E. Nelson. “Testing mu = 0 on the lattice.” In: *Journal of High Energy Physics* 1999.11 (Nov. 1999), p. 027. DOI: 10.1088/1126-6708/1999/11/027. URL: https://dx.doi.org/10.1088/1126-6708/1999/11/027.

[64] CAST Collaboration et al. “New CAST limit on the axion–photon interaction.” In: *Nature Physics* 13 (2017), pp. 584–590. URL: https://www.nature.com/articles/nphys4109.

[65] The IAXO collaboration et al. “Conceptual Design of Babyiaxo, the Intermediate Stage Towards the International Axion Observatory.” In: *Journal of High Energy Physics* 2021.5 (2021), p. 137. DOI: 10.1007/jhep05(2021)137. URL: http://dx.doi.org/10.1007/JHEP05(2021)137.

[66] Glen Cowan. *Statistical data analysis*. Oxford university press, 1998.

[67] William W. Craig et al. “Fabrication of the NuSTAR flight optics.” In: *Optics for EUV, X-Ray, and Gamma-Ray Astronomy V*. Ed. by Stephen L. O’Dell and Giovanni Pareschi. Vol. 8147. International Society for Optics and Photonics. SPIE, 2011, pp. 151–164. DOI: 10.1117/12.895278. URL: https://doi.org/10.1117/12.895278.

[68] Harald Cramér. “On the Composition of Elementary Errors.” In: *Scandinavian Actuarial Journal* 1928.1 (1928), pp. 13–74. DOI: 10.1080/03461238.1928.10416862. URL: http://dx.doi.org/10.1080/03461238.1928.10416862.

[69] R.J. Crewther et al. “Chiral estimate of the electric dipole moment of the neutron in quantum chromodynamics.” In: *Physics Letters B* 88.1 (1979), pp. 123–127. ISSN: 0370-2693. DOI: http://dx.doi.org/10.1016/0370-2693(79)90128-X. URL: http://www.sciencedirect.com/science/article/pii/037026937990128X.

[70] R.J. Crewther et al. “Erratum: Chiral estimate of the electric dipole moment of the neutron in quantum chromodynamics.” In: *Physics Letters B* 91.3 (1980), pp. 487–. ISSN: 0370-2693. DOI: http://dx.doi.org/10.1016/0370-2693(80)91025-4. URL: http://www.sciencedirect.com/science/article/pii/0370269380910254.

[71] S. Arguedas Cuendis et al. “First Results on the Search for Chameleons With the Kwisp Detector At Cast.” In: *Physics of the Dark Universe* 26 (2019), p. 100367. DOI: 10.1016/j.dark.2019.100367. URL: http://dx.doi.org/10.1016/j.dark.2019.100367.

[72] Sergio Arguedas Cuendis. “Dark Matter Axion Search Using Novel Rf Resonant Cavity Geometries in the Cast Experiment.” In: (2021). DOI: 10.25365/THESIS.70391. URL: https://utheses.univie.ac.at/detail/60489.

[73] Theopisti Dafni et al. “Weighing the solar axion.” In: *Physical Review D* 99 (Feb. 2019). DOI: 10.1103/PhysRevD.99.035037.

[74] John Dalton. “Essay IV. On the expansion of elastic fluids by heat.” In: *Memoirs of the Literary and Philosophical Society of Manchester* 5.2 (1802), pp. 595–602.

[75] Alexander Deisting. “Readout and Analysis of the Induced Ion Signal of an InGrid Detector.” BONN-IB-2014-10. MA thesis. Physikalisches Institut der Universität Bonn, Aug. 2014.

[76] Mitchell T Dennis and Jeremy Sakstein. *Tip of the Red Giant Branch Bounds on the Axion-Electron Coupling Revisited*. 2023. arXiv: 2305.03113 [hep-ph].

[77] Christopher Dessert, Andrew J. Long, and Benjamin R. Safdi. "No Evidence for Axions From *Chandra* Observation of the Magnetic White Dwarf Re J0317-853." In: *Physical Review Letters* 128.7 (2022), p. 071102. DOI: 10.1103/physrevlett.128.071102. URL: http://dx.doi.org/10.1103/PhysRevLett.128.071102.

[78] Christopher Dessert, Andrew J. Long, and Benjamin R. Safdi. "X-Ray Signatures of Axion Conversion in Magnetic White Dwarf Stars." In: *Phys. Rev. Lett.* 123 (6 Aug. 2019), p. 061104. DOI: 10.1103/PhysRevLett.123.061104. URL: https://link.aps.org/doi/10.1103/PhysRevLett.123.061104.

[79] Luca Di Luzio et al. "The landscape of QCD axion models." In: *Physics Reports* 870 (2020). The landscape of QCD axion models, pp. 1–117. ISSN: 0370-1573. DOI: https://doi.org/10.1016/j.physrep.2020.06.002. URL: https://www.sciencedirect.com/science/article/pii/S0370157320302477.

[80] Michael Dine, Patrick Draper, and Guido Festuccia. "Instanton effects in three flavor QCD." In: *Phys. Rev. D* 92 (5 Sept. 2015), p. 054004. DOI: 10.1103/PhysRevD.92.054004. URL: https://link.aps.org/doi/10.1103/PhysRevD.92.054004.

[81] Michael Dine, Willy Fischler, and Mark Srednicki. "A simple solution to the strong CP problem with a harmless axion." In: *Physics Letters B* 104.3 (1981), pp. 199–202. ISSN: 0370-2693. DOI: http://dx.doi.org/10.1016/0370-2693(81)90590-6. URL: http://www.sciencedirect.com/science/article/pii/0370269381905906.

[82] A. Von Engel and L. Marton. *Ionized Gases*. 1965.

[83] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise." In: *kdd*. Vol. 96. 34. 1996, pp. 226–231.

[84] U Fano. "Penetration of Protons, Alpha Particles, and Mesons." In: *Annual Review of Nuclear Science* 13.1 (1963), pp. 1–66. DOI: 10.1146/annurev.ns.13.120163.000245. URL: http://dx.doi.org/10.1146/annurev.ns.13.120163.000245.

[85] G.W. Fraser et al. "Potential Solar Axion Signatures in X-Ray Observations With the Xmm-Newton Observatory." In: *Monthly Notices of the Royal Astronomical Society* 445.2 (2014), pp. 2146–2168. DOI: 10.1093/mnras/stu1865. URL: http://dx.doi.org/10.1093/mnras/stu1865.

[86] Peter Friedrich. “ABRIXAS: an imaging x-ray survey in the 0.5- to 10-keV range.” In: *X-Ray Optics, Instruments, and Missions*. Ed. by Richard B. Hoover and Arthur B. C. Walker II. Vol. 3444. International Society for Optics and Photonics. SPIE, 1998, pp. 342–349. URL: https://doi.org/10.1117/12.331248.

[87] J.M. Gablonsky and C.T. Kelley. “A Locally-Biased form of the DIRECT Algorithm.” In: *Journal of Global Optimization* 21.1 (2001), pp. 27–37. DOI: 10.1023/a:1017930332101. URL: http://dx.doi.org/10.1023/A:1017930332101.

[88] Thomas K Gaisser, Ralph Engel, and Elisa Resconi. *Cosmic rays and particle physics*. Cambridge University Press, 2016.

[89] *Garfield++*. URL: https://garfieldpp.web.cern.ch/garfieldpp/.

[90] Shuailiang Ge et al. “X-Ray Annual Modulation Observed By XMM-Newton and Axion Quark Nugget Dark Matter.” In: *Physics of the Dark Universe* 36 (2022), p. 101031. DOI: 10.1016/j.dark.2022.101031. URL: http://dx.doi.org/10.1016/j.dark.2022.101031.

[91] Y. Giomataris. “Development and prospects of the new gaseous detector “Micromegas”.” In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 419.2 (1998), pp. 239–250. ISSN: 0168-9002. DOI: https://doi.org/10.1016/S0168-9002(98)00865-1. URL: https://www.sciencedirect.com/science/article/pii/S0168900298008651.

[92] Y. Giomataris et al. “MICROMEGAS: a high-granularity position-sensitive gaseous detector for high particle-flux environments.” In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 376.1 (1996), pp. 29–35. ISSN: 0168-9002. DOI: https://doi.org/10.1016/0168-9002(96)00175-1. URL: https://www.sciencedirect.com/science/article/pii/0168900296001751.

[93] GNU Project. *GCC, the GNU Compiler Collection*. 2022. URL: https://gcc.gnu.org.

[94] Particle Data Group et al. “Review of Particle Physics.” In: *Progress of Theoretical and Experimental Physics* 2022.8 (Aug. 2022), p. 083C01. ISSN: 2050-3911. DOI: 10.1093/ptep/ptac097. eprint: https://academic.oup.com/ptep/article-pdf/2022/8/083C01/49175539/ptac097.pdf. URL: https://doi.org/10.1093/ptep/ptac097.

[95] Markus Gruber. *GridPix X-ray simulation*. Aug. 2023. URL: https://github.com/GasDet-Bonn/GridPix_xray_simulation.

[96] Markus Gruber. “Study of ionisation and amplification processes in GridPix detectors.” MA thesis. Physikalisches Institut der Universität Bonn, Dec. 2018.

[97] Markus Gruber et al. *tpx-daq*. Version 0.9.0. Version relates to latest at time of writing. 2023. URL: https://github.com/GasDet-Bonn/tpx3-daq.

[98] R. Gunnink. "An Algorithm for Fitting Lorentzian-Broadened, K-Series X-Ray Peaks of the Heavy Elements." In: *Nuclear Instruments and Methods* 143.1 (1977), pp. 145–149. DOI: 10.1016/0029-554x(77)90343-3. URL: http://dx.doi.org/10.1016/0029-554X(77)90343-3.

[99] Charles J. Hailey et al. "The Nuclear Spectroscopic Telescope Array (NuSTAR): optics overview and current status." In: *Space Telescopes and Instrumentation 2010: Ultraviolet to Gamma Ray*. Ed. by Monique Arnaud, Stephen S. Murray, and Tadayuki Takahashi. Vol. 7732. International Society for Optics and Photonics. SPIE, 2010, pp. 197–209. DOI: 10.1117/12.857654. URL: https://doi.org/10.1117/12.857654.

[100] Eric Haines and Tomas Akenine-Möller, eds. *Ray Tracing Gems*. http://raytracinggems.com. Apress, 2019. DOI: https://doi.org/10.1007/978-1-4842-4427-2.

[101] Fiona A. Harrison et al. "Development of the HEFT and NuSTAR focusing telescopes." In: *Focusing Telescopes in Nuclear Astrophysics*. Ed. by Peter von Ballmoos. Dordrecht: Springer Netherlands, 2006, pp. 131–137. ISBN: 978-1-4020-5304-7. DOI: 10.1007/978-1-4020-5304-7_15. URL: https://doi.org/10.1007/978-1-4020-5304-7_15.

[102] Fiona A. Harrison et al. "THENUCLEAR SPECTROSCOPIC TELESCOPE ARRAY(NuSTAR) HIGH-ENERGY X-RAY MISSION." In: *The Astrophysical Journal* 770.2 (May 2013), p. 103. DOI: 10.1088/0004-637x/770/2/103. URL: https://doi.org/10.1088/0004-637x/770/2/103.

[103] W. K. Hastings. "Monte Carlo Sampling Methods Using Markov Chains and Their Applications." In: *Biometrika* 57.1 (1970), pp. 97–109. DOI: 10.1093/biomet/57.1.97. URL: http://dx.doi.org/10.1093/biomet/57.1.97.

[104] J. Heckel and W. Scholz. "Description of Low-energy Peak Distortion Observed in X-ray Spectrometry With Si(Li) Detectors." In: *X-Ray Spectrometry* 16.4 (1987), pp. 181–185. DOI: 10.1002/xrs.1300160409. URL: http://dx.doi.org/10.1002/xrs.1300160409.

[105] Burton L Henke, Eric M Gullikson, and John C Davis. "X-ray interactions: photoabsorption, scattering, transmission, and reflection at E= 50-30,000 eV, Z= 1-92." In: *Atomic data and nuclear data tables* 54.2 (1993), pp. 181–342. URL: https://henke.lbl.gov/optical_constants/.

[106] H. J. Hilke and W. Riegler. "Gaseous Detectors." In: *Particle Physics Reference Library: Volume 2: Detectors for Particles and Radiation*. Ed. by Christian W. Fabjan and Herwig Schopper. Cham: Springer International Publishing, 2020, pp. 91–136. ISBN: 978-3-030-35318-6. DOI: 10.1007/978-3-030-35318-6_4. URL: https://doi.org/10.1007/978-3-030-35318-6_4.

[107] Sebastian Hoof, Joerg Jaeckel, and Lennert J. Thormaehlen. “Quantifying uncertainties in the solar axion flux and their impact on determining axion model parameters.” In: *Journal of Cosmology and Astroparticle Physics* 2021.09 (Sept. 2021), p. 006. DOI: 10.1088/1475-7516/2021/09/006. URL: https://dx.doi.org/10.1088/1475-7516/2021/09/006.

[108] Sebastian Hoof and Lennert Thormaehlen. *SolarAxionFlux. A C++/Python library for calculating the solar axion flux for different solar models and opacity codes.* Available on Github at https://github.com/sebhoof/SolarAxionFlux. [Github repository]. 2021.

[109] Gerard ’t Hooft. *The Glorious Days of Physics - Renormalization of Gauge theories*. 1999. arXiv: hep-th/9812203 [hep-th].

[110] Gerard’t Hooft. “How instantons solve the U (1) problem.” In: *Physics Reports* 142.6 (1986), pp. 357–387. ISSN: 0370-1573.

[111] T. C. Huang and G. Lim. “Profile Analysis of Experimental X-ray Fluorescence Spectra.” In: *X-Ray Spectrometry* 15.4 (1986), pp. 245–250. DOI: 10.1002/xrs.1300150406. URL: http://dx.doi.org/10.1002/xrs.1300150406.

[112] JH Hubbell and SM Seltzer. “NIST standard reference database 126.” In: *Gaithersburg, MD: National Institute of Standards and Technology* (1996). URL: https://www.nist.gov/pml/x-ray-mass-attenuation-coefficients.

[113] DG Hummer and Dimitri Mihalas. “The equation of state for stellar envelopes. I-an occupation probability formalism for the truncation of internal partition functions.” In: *Astrophysical Journal, Part 1 (ISSN 0004-637X), vol. 331, Aug. 15, 1988, p. 794-814. Research supported by the Alexander von Humboldt Stiftung.* 331 (1988), pp. 794–814.

[114] J. D. Hunter. “Matplotlib: A 2D graphics environment.” In: *Computing in Science & Engineering* 9.3 (2007), pp. 90–95. DOI: 10.1109/MCSE.2007.55.

[115] David S. Immel, Michael F. Cohen, and Donald P. Greenberg. “A radiosity method for non-diffuse environments.” In: *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*. Aug. 1986. DOI: 10.1145/15922.15901. URL: http://dx.doi.org/10.1145/15922.15901.

[116] Y. Inoue et al. “Search for solar axions with mass around 1 eV using coherent conversion of axions into photons.” In: *Physics Letters B* 668.2 (2008), pp. 93–97. ISSN: 0370-2693. DOI: https://doi.org/10.1016/j.physletb.2008.08.020. URL: https://www.sciencedirect.com/science/article/pii/S0370269308010137.

[117] Yoshizumi Inoue et al. “Search for sub-electronvolt solar axions using coherent conversion of axions into photons in magnetic field and gas helium.” In: *Physics Letters B* 536.1 (2002), pp. 18–23. ISSN: 0370-2693. DOI: https://doi.org/10.1016/S0370-2693(02)01822-1. URL: https://www.sciencedirect.com/science/article/pii/S0370269302018221.

[118] I.G Irastorza et al. “Towards a new generation axion helioscope.” In: *Journal of Cosmology and Astroparticle Physics* 2011.06 (June 2011), pp. 013–013. DOI: 10.1088/1475-7516/2011/06/013. URL: https://doi.org/10.1088/1475-7516/2011/06/013.

[119] Igor García Irastorza. “An introduction to axions and their detection.” In: *SciPost Phys. Lect. Notes* (2022), p. 45. DOI: 10.21468/SciPostPhysLectNotes.45. URL: https://scipost.org/10.21468/SciPostPhysLectNotes.45.

[120] Anders Clemen Jakobsen. “X-ray optics in new instruments for astro- and astroparticle physics.” English. PhD thesis. 2015. ISBN: 978-87-91694-28-8.

[121] *JLP Horizons*. URL: https://ssd.jpl.nasa.gov/horizons/ (visited on 08/05/2023).

[122] Steven G. Johnson. *The NLopt nonlinear-optimization package*. https://github.com/stevengj/nlopt. 2007.

[123] D. R. Jones, C. D. Perttunen, and B. E. Stuckman. “Lipschitzian Optimization Without the Lipschitz Constant.” In: *Journal of Optimization Theory and Applications* 79.1 (1993), pp. 157–181. DOI: 10.1007/bf00941892. URL: http://dx.doi.org/10.1007/BF00941892.

[124] Thomas Junk. “Confidence Level Computation for Combining Searches With Small Statistics.” In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 434.2-3 (1999), pp. 435–443. DOI: 10.1016/s0168-9002(99)00498-2. URL: http://dx.doi.org/10.1016/S0168-9002(99)00498-2.

[125] James T. Kajiya. “The rendering equation.” In: *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*. Aug. 1986. DOI: 10.1145/15922.15902. URL: http://dx.doi.org/10.1145/15922.15902.

[126] David B. Kaplan and Aneesh V. Manohar. “Current-Mass Ratios of the Light Quarks.” In: *Phys. Rev. Lett.* 56 (19 May 1986), pp. 2004–2007. DOI: 10.1103/PhysRevLett.56.2004. URL: https://link.aps.org/doi/10.1103/PhysRevLett.56.2004.

[127] Jihn E. Kim. “Weak-Interaction Singlet and Strong CP Invariance.” In: *Phys. Rev. Lett.* 43 (2 July 1979), pp. 103–107. DOI: 10.1103/PhysRevLett.43.103. URL: http://link.aps.org/doi/10.1103/PhysRevLett.43.103.

[128] Jihn E. Kim and Gianpaolo Carosi. “Axions and the strong *CP* problem.” In: *Rev. Mod. Phys.* 82 (1 Mar. 2010), pp. 557–601. DOI: 10.1103/RevModPhys.82.557. URL: https://link.aps.org/doi/10.1103/RevModPhys.82.557.

[129] Diederik P. Kingma and Jimmy Ba. “Adam: a Method for Stochastic Optimization.” In: (2014). DOI: 10.48550/ARXIV.1412.6980. URL: https://arxiv.org/abs/1412.6980.

[130] Erik Bergbäck Knudsen et al. “*McXtrace*: a Monte Carlo Software Package for Simulating X-Ray Optics, Beamlines and Experiments.” In: *Journal of Applied Crystallography* 46.3 (2013), pp. 679–696. DOI: 10.1107/s0021889813007991. URL: http://dx.doi.org/10.1107/S0021889813007991.

[131] Jason E. Koglin et al. “NuSTAR hard x-ray optics design and performance.” In: *Optics for EUV, X-Ray, and Gamma-Ray Astronomy IV*. Ed. by Stephen L. O’Dell and Giovanni Pareschi. Vol. 7437. International Society for Optics and Photonics. SPIE, 2009, pp. 107–114. URL: https://doi.org/10.1117/12.826724.

[132] Hermann Kolanoski and Norbert Wermes. *Particle Detectors: Fundamentals and Applications*. Oxford University Press, USA, 2020.

[133] C. Krieger et al. “Energy Dependent Features of X-Ray Signals in a Gridpix Detector.” In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 893 (2018), pp. 26–34. DOI: 10.1016/j.nima.2018.02.109. URL: http://dx.doi.org/10.1016/j.nima.2018.02.109.

[134] Christoph Krieger. “Search for solar chameleons with an InGrid based X-ray detector at the CAST experiment.” PhD thesis. 2018.

[135] M Kuster et al. “The x-ray telescope of CAST.” In: *New Journal of Physics* 9.6 (2007), p. 169. URL: http://stacks.iop.org/1367-2630/9/i=6/a=169.

[136] Markus Kuster, Georg Raffelt, and Berta Beltrán. *Axions: Theory, cosmology, and experimental searches*. Vol. 741. Springer, 2007.

[137] Johann Heinrich Lambert. *Photometria sive de mensura et gradibus luminis, colorum et umbrae*. sumptibus vidvae E. Klett, typis CP Detleffsen, 1760. URL: https://archive.org/details/TO0E039861_TO0324_PNI-2733_000000.

[138] Chris Lattner and Vikram Adve. “LLVM: A Compilation Framework for Lifelong Program Analysis and Transformation.” In: San Jose, CA, USA, Mar. 2004, pp. 75–88.

[139] D. M. Lazarus et al. “Search for solar axions.” In: *Phys. Rev. Lett.* 69 (16 Oct. 1992), pp. 2333–2336. DOI: 10.1103/PhysRevLett.69.2333. URL: https://link.aps.org/doi/10.1103/PhysRevLett.69.2333.

[140] Luca Lista. *Statistical Methods for Data Analysis*. Lecture Notes in Physics. Springer International Publishing, 2023. DOI: 10.1007/978-3-031-19934-9. URL: http://dx.doi.org/10.1007/978-3-031-19934-9.

[141] X. Llopart et al. "Medipix2: A 64-k pixel readout chip with 55- mu;m square elements working in single photon counting mode." In: *IEEE Transactions on Nuclear Science* 49.5 (Oct. 2002), pp. 2279–2283. ISSN: 0018-9499. DOI: 10.1109/TNS.2002.803788.

[142] X. Llopart et al. "Timepix, a 65k programmable pixel readout chip for arrival time, energy and/or photon counting measurements." In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 581.1 (2007). VCI 2007, pp. 485–494. ISSN: 0168-9002. DOI: https://doi.org/10.1016/j.nima.2007.08.079. URL: https://www.sciencedirect.com/science/article/pii/S0168900207017020.

[143] Xavier Llopart and Tuomas Poikela. "Timepix Manual v1.0." In: *CERN, Switzerland* (2006).

[144] Xavier Llopart and Tuomas Poikela. "Timepix3 Manual v2.0." In: *CERN, Switzerland* (2015).

[145] Xavier Llopart Cudie. "Design and Characterization of 64K Pixels Chips Working in Single Photon Processing Mode." Presented on 25 May 2007. PhD thesis. 2007. URL: https://cds.cern.ch/record/1056683.

[146] Michael Lupberger. "The Pixel-TPC: A feasibility study." PhD thesis. Universitäts-und Landesbibliothek Bonn, 2016.

[147] Luca Di Luzio et al. "Probing the Axion-Nucleon Coupling With the Next Generation of Axion Helioscopes." In: *The European Physical Journal C* 82.2 (2022), p. 120. DOI: 10.1140/epjc/s10052-022-10061-1. URL: http://dx.doi.org/10.1140/epjc/s10052-022-10061-1.

[148] Luca Di Luzio et al. "Solar Axions Cannot Explain the Xenon1t Excess." In: *Physical Review Letters* 125.13 (2020), p. 131804. DOI: 10.1103/physrevlett.125.131804. URL: http://dx.doi.org/10.1103/PhysRevLett.125.131804.

[149] Marios Maroudas. "Signals for invisible matter from solar-terrestrial observations." PhD thesis. 2022. URL: http://dx.doi.org/10.12681/eadd/51922.

[150] Emi Masaki, Arata Aoki, and Jiro Soda. "Photon-Axion Conversion, Magnetic Field Configuration, and Polarization of Photons." In: *Physical Review D* 96.4 (2017), p. 043519. DOI: 10.1103/physrevd.96.043519. URL: http://dx.doi.org/10.1103/PhysRevD.96.043519.

[151] A. Álvarez Melcón et al. "First Results of the Cast-Rades Haloscope Search for Axions At 34.67 μeV." In: *Journal of High Energy Physics* 2021.10 (2021), p. 75. DOI: 10.1007/jhep10(2021)075. URL: http://dx.doi.org/10.1007/JHEP10(2021)075.

[152] Nicholas Metropolis et al. “Equation of State Calculations By Fast Computing Machines.” In: *The Journal of Chemical Physics* 21.6 (1953), pp. 1087–1092. DOI: 10.1063/1.1699114. URL: http://dx.doi.org/10.1063/1.1699114.

[153] Karnig O. Mikaelian. “Astrophysical implications of new light Higgs bosons.” In: *Phys. Rev. D* 18 (10 Nov. 1978), pp. 3605–3609. DOI: 10.1103/PhysRevD.18.3605. URL: https://link.aps.org/doi/10.1103/PhysRevD.18.3605.

[154] Richard von Mises. *Wahrscheinlichkeit Statistik und Wahrheit*. Springer Berlin Heidelberg, 1936. DOI: 10.1007/978-3-662-41863-5. URL: http://dx.doi.org/10.1007/978-3-662-41863-5.

[155] Shigetaka Moriyama et al. “Direct search for solar axions by using strong magnetic field and X-ray detectors.” In: *Physics Letters B* 434.1 (1998), pp. 147–152. ISSN: 0370-2693. DOI: https://doi.org/10.1016/S0370-2693(98)00766-7. URL: https://www.sciencedirect.com/science/article/pii/S0370269398007667.

[156] *MT_RAYOR - A raytracing system for X-ray telescopes*. Version 4.6.5. URL: https://ftp.spacecenter.dk/pub/njw/MT_RAYOR/mt_rayor_man4.pdf.

[157] Ciaran O’HARE. *cajohare/AxionLimits: AxionLimits*. Version v1.0. July 2020. DOI: 10.5281/zenodo.3932430. URL: https://doi.org/10.5281/zenodo.3932430.

[158] Johanna von Oy. “Monte Carlo based ray tracing for the search of solar axions with CAST.” MA thesis. Universität zu Köln, June 2020.

[159] Johanna von Oy. *SolarAxionRayTracing*. 2023. URL: https://github.com/jovoy/SolarAxionRayTracing.

[160] S. Pancheshnyi et al. “The Lxcat Project: Electron Scattering Cross Sections and Swarm Parameters for Low Temperature Plasma Modeling.” In: *Chemical Physics* 398 (2012), pp. 148–153. DOI: 10.1016/j.chemphys.2011.04.020. URL: http://dx.doi.org/10.1016/j.chemphys.2011.04.020.

[161] Friedrich Paschen. “Ueber Die Zum Funkenübergang in Luft, Wasserstoff Und Kohlensäure Bei Verschiedenen Drucken Erforderliche Potentialdifferenz.” In: *Annalen der Physik* 273.5 (1889), pp. 69–96. DOI: 10.1002/andp.18892730505. URL: http://dx.doi.org/10.1002/andp.18892730505.

[162] Adam Paszke et al. “PyTorch: An Imperative Style, High-Performance Deep Learning Library.” In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[163] R. D. Peccei and Helen R. Quinn. “CP Conservation in the Presence of Pseudoparticles.” In: *Phys. Rev. Lett.* 38 (25 June 1977), pp. 1440–1443. DOI: 10.1103/PhysRevLett.38.1440. URL: http://link.aps.org/doi/10.1103/PhysRevLett.38.1440.

[164] R. D. Peccei and Helen R. Quinn. “Constraints imposed by CP conservation in the presence of pseudoparticles.” In: *Phys. Rev. D* 16 (6 Sept. 1977), pp. 1791–1797. DOI: 10.1103/PhysRevD.16.1791. URL: http://link.aps.org/doi/10.1103/PhysRevD.16.1791.

[165] Roberto D. Peccei. “The Strong CP Problem and Axions.” In: *Axions: Theory, Cosmology, and Experimental Searches*. Ed. by Markus Kuster, Georg Raffelt, and Berta Beltrán. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 3–17. ISBN: 978-3-540-73518-2. DOI: 10.1007/978-3-540-73518-2_1. URL: https://doi.org/10.1007/978-3-540-73518-2_1.

[166] Roberto D. Peccei. “The Strong CP Problem and Axions.” In: *Lecture Notes in Physics*. Springer Berlin Heidelberg, 2008, pp. 3–17. DOI: 10.1007/978-3-540-73518-2_1. URL: https://doi.org/10.1007%2F978-3-540-73518-2_1.

[167] M.J.M. Pelgrom, A.C.J. Duinmaijer, and A.P.G. Welbers. “Matching properties of MOS transistors.” In: *IEEE Journal of Solid-State Circuits* 24.5 (1989), pp. 1433–1439. DOI: 10.1109/JSSC.1989.572629.

[168] J. M. Pendlebury et al. “Revised experimental upper limit on the electric dipole moment of the neutron.” In: *Phys. Rev. D* 92 (9 Nov. 2015), p. 092003. DOI: 10.1103/PhysRevD.92.092003. URL: http://link.aps.org/doi/10.1103/PhysRevD.92.092003.

[169] Robert Petre and Peter J. Serlemitsos. “Conical imaging mirrors for high-speed x-ray telescopes.” In: *Appl. Opt.* 24.12 (June 1985), pp. 1833–1837. DOI: 10.1364/AO.24.001833. URL: https://opg.optica.org/ao/abstract.cfm?URI=ao-24-12-1833.

[170] Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically based rendering: From theory to implementation*. Morgan Kaufmann, 2016.

[171] Leanne C. Pitchford et al. “Lxcat: an Open-access, Web-based Platform for Data Needed for Modeling Low Temperature Plasmas.” In: *Plasma Processes and Polymers* 14.1-2 (2016). DOI: 10.1002/ppap.201600098. URL: http://dx.doi.org/10.1002/ppap.201600098.

[172] T Poikela et al. “Timepix3: a 65K channel hybrid pixel readout chip with simultaneous ToA/ToT and sparse readout.” In: *Journal of Instrumentation* 9.05 (May 2014), p. C05013. DOI: 10.1088/1748-0221/9/05/C05013. URL: https://dx.doi.org/10.1088/1748-0221/9/05/C05013.

[173] Andrius Poškus. “Monte Carlo estimation of average energy required to produce an ion pair in noble gases by electrons with energies from 1 keV to 100 MeV.” In: *Journal of Nuclear Science and Technology* 52.5 (2015), pp. 675–686. DOI: 10.1080/00223131.2014.974710.

eprint: https://doi.org/10.1080/00223131.2014.974710. URL: https://doi.org/10.1080/00223131.2014.974710.

[174] Georg Raffelt and Leo Stodolsky. "Mixing of the Photon With Low-Mass Particles." In: *Physical Review D* 37.5 (1988), pp. 1237–1249. DOI: 10.1103/physrevd.37.1237. URL: http://dx.doi.org/10.1103/PhysRevD.37.1237.

[175] Georg G Raffelt. *Stars as laboratories for fundamental physics: The astrophysics of neutrinos, axions, and other weakly interacting particles*. University of Chicago press, 1996.

[176] Georg G. Raffelt. "Astrophysical Axion Bounds Diminished By Screening Effects." In: *Physical Review D* 33.4 (1986), pp. 897–909. DOI: 10.1103/physrevd.33.897. URL: http://dx.doi.org/10.1103/PhysRevD.33.897.

[177] Georg G. Raffelt. "Plasmon Decay Into Low-Mass Bosons in Stars." In: *Physical Review D* 37.6 (1988), pp. 1356–1359. DOI: 10.1103/physrevd.37.1356. URL: http://dx.doi.org/10.1103/PhysRevD.37.1356.

[178] Javier Redondo. "Solar axion flux from the axion-electron coupling." In: *Journal of Cosmology and Astroparticle Physics* 2013.12 (Dec. 2013), pp. 008–008. DOI: 10.1088/1475-7516/2013/12/008. URL: https://doi.org/10.1088/1475-7516/2013/12/008.

[179] G. Richter et al. "Abrixas." In: *Imaging in High Energy Astronomy*. Ed. by L. Bassani and G. Di Cocco. Dordrecht: Springer Netherlands, 1995, pp. 159–162. ISBN: 978-94-011-0407-4. DOI: 10.1007/978-94-011-0407-4_21. URL: https://doi.org/10.1007/978-94-011-0407-4_21.

[180] A. Ringwald. "Discovery Potential for Axions in Hamburg." In: (2023). DOI: 10.48550/ARXIV.2306.08978. URL: https://arxiv.org/abs/2306.08978.

[181] B.L. Roberts, R.A.J. Riddle, and G.T.A. Squier. "Measurement of Lorentzian Linewidths." In: *Nuclear Instruments and Methods* 130.2 (1975), pp. 559–563. DOI: 10.1016/0029-554x(75)90058-0. URL: http://dx.doi.org/10.1016/0029-554X(75)90058-0.

[182] W. Romberg. "Vereinfachte numerische Integration." In: *Norske Vid. Selsk. Forh.* 28 (1955), pp. 30–36. URL: https://cir.nii.ac.jp/crid/1573950399004224384.

[183] M. Roncadelli and F. Tavecchio. "No Axions From the Sun." In: *Monthly Notices of the Royal Astronomical Society: Letters* 450.1 (2015), pp. L26–L28. DOI: 10.1093/mnrasl/slv040. URL: http://dx.doi.org/10.1093/mnrasl/slv040.

[184] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. "Learning Representations By Back-Propagating Errors." In: *Nature* 323.6088 (1986), pp. 533–536. DOI: 10.1038/323533a0. URL: http://dx.doi.org/10.1038/323533a0.

[185] Andreas Rump. *Nim programming language*. 2022. URL: https://nim-lang.org.

[186] Fabio Sauli. *Gaseous radiation detectors: fundamentals and applications*. 36. Cambridge University Press, 2014.

[187] Lucian Scharenberg. "Studies on the Thermal Behaviour of GridPix-Based Detectors." MA thesis. Physikalisches Institut der Universität Bonn, Feb. 2019.

[188] Tobias Schiffer. "Upcoming PhD thesis about a GridPix3 detector." PhD thesis. 2024.

[189] Hendrik Schmick. "Characterization of an improved GridPix detector for solar axion search at CAST." MA thesis. Physikalisches Institut der Universität Bonn, July 2019.

[190] Juergen Schmidhuber. "Annotated History of Modern Ai and Deep Learning." In: (2022). DOI: 10.48550/ARXIV.2212.11279. URL: https://arxiv.org/abs/2212.11279.

[191] Sebastian Schmidt. *Data archive for 7-GridPix 'Septemboard' detector taken at CAST (2017/18)*. Zenodo, Feb. 2024. DOI: 10.5281/zenodo.10521887. URL: https://doi.org/10.5281/zenodo.10521887.

[192] Sebastian Schmidt. "Search for particles beyond the SM using an InGrid detector at CAST." MA thesis. Physikalisches Institut der Universität Bonn, Nov. 2016.

[193] Sebastian Schmidt. *TrAXer - an interactive real-time X-ray raytracer*. 2023. URL: https://github.com/Vindaar/TrAXer.

[194] Sebastian Michael Schmidt. *TimepixAnalysis*. 2022. URL: https://github.com/Vindaar/TimepixAnalysis.

[195] Sebastian Michael Schmidt and SciNim contributors. *xrayAttenuation*. Version 0.1.0. Sept. 2022. URL: https://github.com/SciNim/xrayAttenuation.

[196] Jannes Schmitz. "Bau von Szintillationsvetodetektoren für das CAST-Experiment." bscthesis. Physikalisches Institut der Universität Bonn, Aug. 2017.

[197] Erich Schubert et al. "DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN." In: *ACM Trans. Database Syst.* 42.3 (July 2017). ISSN: 0362-5915. DOI: 10.1145/3068335. URL: https://doi.org/10.1145/3068335.

[198] SciNim contributors. *Flambeau*. Version 0.0.3. 2023. URL: https://github.com/SciNim/Flambeau.

[199] *SDL - Simple DirectMedia Layer*. Version 2. URL: https://www.libsdl.org/.

[200] MJ Seaton. "Atomic data for opacity calculations. I. General description." In: *Journal of Physics B: Atomic and Molecular Physics* 20.23 (1987), p. 6363.

[201] MJ Seaton. "MNRAS Letters 362, Issue 1. pp." In: *L1-L3* (2005).

[202] MJ Seaton et al. “Opacities for stellar envelopes.” In: *Monthly Notices of the Royal Astronomical Society* 266.4 (1994), pp. 805–828.

[203] Aldo M. Serenelli et al. “New Solar Composition: the Problem With Solar Models Revisited.” In: *The Astrophysical Journal* 705.2 (2009), pp. L123–L127. DOI: 10.1088/0004-637x/705/2/l123. URL: http://dx.doi.org/10.1088/0004-637X/705/2/L123.

[204] M.A. Shifman, A.I. Vainshtein, and V.I. Zakharov. “Can confinement ensure natural CP invariance of strong interactions?” In: *Nuclear Physics B* 166.3 (1980), pp. 493–506. ISSN: 0550-3213. DOI: http://dx.doi.org/10.1016/0550-3213(80)90209-6. URL: http://www.sciencedirect.com/science/article/pii/0550321380902096.

[205] Peter Shirley. *Ray Tracing in One Weekend*. Dec. 2020. URL: https://raytracing.github.io/books/RayTracingInOneWeekend.html (visited on 10/18/2022).

[206] Prashant Shukla and Sundaresh Sankrith. “Energy and angular distributions of atmospheric muons at the Earth.” In: *International Journal of Modern Physics A* 33.30 (2018), p. 1850175. DOI: 10.1142/S0217751X18501750. URL: https://doi.org/10.1142/S0217751X18501750.

[207] P. Sikivie. “Detection rates for “invisible”-axion searches.” In: *Phys. Rev. D* 32 (11 Dec. 1985), pp. 2988–2991. DOI: 10.1103/PhysRevD.32.2988. URL: https://link.aps.org/doi/10.1103/PhysRevD.32.2988.

[208] P. Sikivie. “Experimental Tests of the ”Invisible” Axion.” In: *Phys. Rev. Lett.* 51 (16 Oct. 1983), pp. 1415–1417. DOI: 10.1103/PhysRevLett.51.1415. URL: https://link.aps.org/doi/10.1103/PhysRevLett.51.1415.

[209] Pierre Sikivie. “Invisible axion search methods.” In: *Rev. Mod. Phys.* 93 (1 Feb. 2021), p. 015004. DOI: 10.1103/RevModPhys.93.015004. URL: https://link.aps.org/doi/10.1103/RevModPhys.93.015004.

[210] Mark Srednicki. “Axion couplings to matter: (I). CP-conserving parts.” In: *Nuclear Physics B* 260.3 (1985), pp. 689–700. ISSN: 0550-3213. DOI: https://doi.org/10.1016/0550-3213(85)90054-9. URL: https://www.sciencedirect.com/science/article/pii/0550321385900549.

[211] O. Straniero et al. “The Rgb Tip of Galactic Globular Clusters and the Revision of the Axion-Electron Coupling Bound.” In: *Astronomy & Astrophysics* 644 (2020), A166. DOI: 10.1051/0004-6361/202038775. URL: http://dx.doi.org/10.1051/0004-6361/202038775.

[212] L. Strüder et al. “The European Photon Imaging Camera on Xmm-Newton: the Pn-Ccd Camera.” In: *Astronomy & Astrophysics* 365.1 (2001), pp. L18–L26. DOI: 10.1051/0004-6361:20000066. URL: http://dx.doi.org/10.1051/0004-6361:20000066.

[213] Opacity Team. *The opacity project*. 1995.

[214] The HDF Group. *Hierarchical Data Format, version 5*. https://www.hdfgroup.org/HDF5/. 1997-2022. (Visited on 11/12/2022).

[215] *TOS - Timepix Operating Software*. URL: https://github.com/Vindaar/TOS.

[216] Turner, M. J. L. et al. "The European Photon Imaging Camera on XMM-Newton: The MOS cameras." In: *A&A* 365.1 (2001), pp. L27–L35. DOI: 10.1051/0004-6361:20000087. URL: https://doi.org/10.1051/0004-6361:20000087.

[217] Eric Veach. "Robust Monte Carlo methods for light transport simulation." PhD thesis. 1998.

[218] J. K. Vogel et al. "IAXO - The International Axion Observatory." In: (2013). DOI: 10.48550/ARXIV.1302.3273. URL: https://arxiv.org/abs/1302.3273.

[219] Voltaire. *Micromégas*. 1752.

[220] T. P. Waterhouse. *An Introduction to Chameleon Gravity*. 2006. DOI: 10.48550/ARXIV.ASTRO-PH/0611816. arXiv: astro-ph/0611816 [astro-ph]. URL: https://arxiv.org/abs/astro-ph/0611816.

[221] Steven Weinberg. "A New Light Boson?" In: *Phys. Rev. Lett.* 40 (4 Jan. 1978), pp. 223–226. DOI: 10.1103/PhysRevLett.40.223. URL: http://link.aps.org/doi/10.1103/PhysRevLett.40.223.

[222] Steven Weinberg. "THE PROBLEM OF MASS." In: *Transactions of the New York Academy of Sciences* 38.1 Series II (1977), pp. 185–201. DOI: https://doi.org/10.1111/j.2164-0947.1977.tb02958.x. eprint: https://nyaspubs.onlinelibrary.wiley.com/doi/pdf/10.1111/j.2164-0947.1977.tb02958.x. URL: https://nyaspubs.onlinelibrary.wiley.com/doi/abs/10.1111/j.2164-0947.1977.tb02958.x.

[223] Steven Weinberg. "The U(1) problem." In: *Phys. Rev. D* 11 (12 June 1975), pp. 3583–3593. DOI: 10.1103/PhysRevD.11.3583. URL: https://link.aps.org/doi/10.1103/PhysRevD.11.3583.

[224] V. Weisskopf and E. P. Wigner. "Berechnung der natürlichen Linienbreite auf Grund der Diracschen Lichttheorie." In: *Part I: Particles and Fields. Part II: Foundations of Quantum Mechanics*. Part I: Particles and Fields. Part II: Foundations of Quantum Mechanics. Springer Berlin Heidelberg, 1997, pp. 30–49. DOI: 10.1007/978-3-662-09203-3_3. URL: http://dx.doi.org/10.1007/978-3-662-09203-3_3.

[225] Turner Whitted. "An improved illumination model for shaded display." In: *Proceedings of the 6th annual conference on Computer graphics and interactive techniques*. Aug. 1979. DOI: 10.1145/800249.807419. URL: http://dx.doi.org/10.1145/800249.807419.

[226] F. Wilczek. “Problem of Strong P and T Invariance in the Presence of Instantons.” In: *Phys. Rev. Lett.* 40 (5 Jan. 1978), pp. 279–282. DOI: 10.1103/PhysRevLett.40.279. URL: http://link.aps.org/doi/10.1103/PhysRevLett.40.279.

[227] Mark D. Wilkinson et al. “The Fair Guiding Principles for Scientific Data Management and Stewardship.” In: *Scientific Data* 3.1 (2016), p. 160018. DOI: 10.1038/sdata.2016.18. URL: http://dx.doi.org/10.1038/sdata.2016.18.

[228] Gwyn P Williams. “X-ray data booklet.” In: *X-ray data booklet* (2001). URL: https://xdb.lbl.gov/xdb-new.pdf.

[229] David L. Windt. “IMD-Software for Modeling the Optical Properties of Multilayer Films.” In: *Computers in Physics* 12.4 (1998), pp. 360–370. DOI: 10.1063/1.168689. URL: http://dx.doi.org/10.1063/1.168689.

[230] Hans Wolter. “Spiegelsysteme streifenden Einfalls als abbildende Optiken für Röntgenstrahlen.” In: *Annalen der Physik* 445.1-2 (1952), pp. 94–114. DOI: https://doi.org/10.1002/andp.19524450108. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/andp.19524450108. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/andp.19524450108.

[231] Kerry Wong. *MCP2210-Library*. May 2013. URL: https://github.com/kerrydwong/MCP2210-Library.

[232] *Yorick - interpreted programming language*. URL: https://github.com/LLNL/yorick.

[233] Andrea Zanzi and Barbara Ricci. “Chameleon Fields and Solar Physics.” In: *Modern Physics Letters A* 30.10 (2015), p. 1550053. DOI: 10.1142/s0217732315500534. URL: http://dx.doi.org/10.1142/S0217732315500534.

[234] A.P. Zhitnitskii. “Possible suppression of axion-hadron interactions.” In: *Sov. J. Nucl. Phys. (Engl. Transl.); (United States)* 31:2 (Feb. 1980).

[235] K. Zioutas et al. “A decommissioned LHC model magnet as an axion telescope.” In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 425.3 (1999), pp. 480–487. ISSN: 0168-9002. DOI: https://doi.org/10.1016/S0168-9002(98)01442-9. URL: https://www.sciencedirect.com/science/article/pii/S0168900298014429.

[236] K. Zioutas et al. “First Results from the CERN Axion Solar Telescope.” In: *Phys. Rev. Lett.* 94 (12 Apr. 2005), p. 121301. DOI: 10.1103/PhysRevLett.94.121301. URL: https://link.aps.org/doi/10.1103/PhysRevLett.94.121301.

[237] P.A. Zyla et al. “Review of Particle Physics.” In: *PTEP* 2020.8 (2020), p. 083C01. DOI: 10.1093/ptep/ptaa104.

A. Data acquisition and detector monitoring Appendix:Detector

Contents

Having introduced the detector used for the data taking in chapter 6, this appendix introduces the data acquisition (DAQ) software for the detector (sec. A.1 and A.2), discusses the data formats used for readout as well as the logging facilities. In the last section A.3 we will discuss the event display tool used to monitor daily data taking at CAST.

A.1. Timepix Operating Firmware - TOF

Starting with the firmware of the detector, the **T**imepix **O**perating **F**irmware (TOF), which runs on the Virtex-6 FPGA, specifically a Xilinx Virtex-6 (V6) ML605 evaluation board. TOF controls the Timepix ASICs of the Septemboard (both the slow control aspects and data taking) as well as coordinates the scintillator signals and FADC trigger. It is a VHDL project, intended to run at a clock frequency of 40 MHz. Communication with the GridPixes is done via two **H**igh-**D**efinition **M**ultimedia **I**nterface (HDMI) cables, while communication with the readout software on the DAQ computer is handled via Ethernet. For a detailed introduction to TOF, see [146] as well as [188].

See sec. B.2 for a few notes on the used firmware versions during data taking.

A.2. Timepix Operating Software - TOS

The **T**imepix **O**perating **S**oftware (TOS) is the computer-side data acquisition software to read out Timepix based detectors. It is an object oriented C++ project, available at [215]. [1] The project needs to be used in conjunction with the **T**imepix **O**perating **F**irmware (TOF), which communicates with TOS via Ethernet. The TOS project started as far back as 2009 by people at the University of Mainz. Next is a short overview over the basic blocks that make up the main logic of the software.

The fully object oriented nature of the project means that there are different classes for the different software pieces:

`Console` A class representing the user facing REPL (Read-Evaluate-Print Loop, an 'interpreter') to control the software

`PC` A class representing the network layer and communication side of the software, sitting between the console and lower layers.

`FPGA` A class representing the functionality required to control the FPGA on the Virtex-6 evaluation board.

`Chip` A class representing each Timepix ASIC and its functionality.

`HFManager` A class unifying the FADC & Wiener HV control unit as they are both controlled via USB, installed in a VME crate. This class contains individual attributes that contain explicit classes for these two devices. The name is shortened for 'High Voltage and FADC Manager'.

`V1729` A class representing the Ortec Flash ADC.

`HV*` Multiple classes representing HV channels, groups and more.

`MCP2210` A class representing the PT1000 temperature sensors installed on the detector via a `MCP2210` micro controller, optionally connected via USB. The actual micro-controllers with attached PT1000s are `MAX31685` models.

Misc there are a few further classes of minor interest to the general functionality of TOS (tab command completion and history, classes to set masks on the chips, etc.)

In general TOS is a fully command line driven software package, with its own REPL (Read-Evaluate-Print Loop; the name for an interactive terminal process, which takes commands that are evaluated

[1] There are unfortunately 2 different versions of TOS, as development diverged for different readout systems. One version is for the Xilinx Virtex-6 (V6) ML605 evaluation board and the other for the **S**calable **R**eadout **S**ystem (SRS). The V6 version can read out only a single detector (with up to 8 Timepix ASICs), but supports readout of an Ortec FADC and controlling a Wiener HV module via VME. The SRS version instead supports neither of these additional features, but supports multiple detectors at the same time. The detector used in this thesis is read out using the Virtex-6 board.

and returns to the terminal). It brings all the expected features one might wish from a REPL, including auto completion, history lookup, emacs style keybindings (based on GNU Readline [2]) and more.

The aforementioned `HFManager` and the temperature sensors are optional pieces that are not required for basic Timepix operation. Their functionality has to be activated via a manual command, `ActivateHFM`. This triggers the USB connection to the VME crate and tries to find the Wiener HV module as well as the Ortec FADC in the crate. Additionally, the temperature sensors are attempted to be found (via a secondary, optional USB connection). If the latter are found a continuous temperature logging begins (see sec. A.2.4).

The HV controls are specific to Wiener HV power supplies. In principle the implemented functionality is a fully featured HV controller that supports all Wiener functionality like grouping different channels to ramp up together, kill channels on a trip and more. Most importantly, it implements a custom, slow HV ramping logic, which keeps the relative potentials constant between channels in a group to avoid tripping a channel.

An example of a typical startup procedure is shown in listing 1, in this case to start a background run. Note that most essential commands in TOS also have shortened names via numbers, due to historic reasons (TOS originally did not have autocompletion or allowed moving the cursor in text input, making typing complex names cumbersome and error prone), which is why many of the inputs are simple numbers.

A.2.1. TOS output data format

When starting a data taking run with TOS, a new directory for the run is created in the `data/runs` subdirectory. The name will be of the form `Run_<run number>_YYMMdd-HH-mm` where the run number is an increasing number based on the last run present in the directory and the suffix is the date and time of day when starting the run. This directory contains the configuration of all DACs for each chip, `fsr<chip number>.txt`, the written configuration matrix for all pixels, `matrix<chip number>.txt` and finally the data files `data<event number>.txt`. If an FADC is used for the readout additional `data<event number>.txt-fadc` files are created, one for each file in which the FADC triggered (sec. A.2.1).

The Timepix data files are stored – for historic reasons – in raw ASCII format. Two different readout modes (with different output formats) are supported. For the following explanation it is assumed the Timepix is used in the ToT (Time-over-Threshold) mode.

full matrix readout reads out the whole Timepix ASIC(s) and writes a single 256x256 pixel matrix as an ASCII file (for each chip). 256 lines, each containing space separated ToT values for each

[2] `https://tiswww.case.edu/php/chet/readline/rltop.html`

```
user@ingrid-DAQ~/ ./TOS
  > 7 # number of chips
  > 4 # preload
> SetChipIDOffset
  > 190
> lf # load FSR values for the chips
  > # return 7 times enter to load default paths
> uma # create a uniform matrix for all chips
  > 1 # Matrix settings
  > 0
  > 1
  > 1
  > 0
> LoadThreshold # load threshold equalisation files
  > 4 # write matrix
  > 3 # read out
  > 3 # 2nd readout to make sure pixels are 'empty'
> ActivateHFM # startup HV & FADC controls
> SetFadcSettings # load the FADC settings
> Run # start a data taking run
  > 1 # run time via # frames
  > 0
  > 0
  > 0
  > 2 # shutter range select
  > 30 # shutter time select (2 + 30 yields ~2.2 s frames)
  > 0 # zero suppression
  > 1 # FADC usage
  > 0 # accept FADC settings
```

Listing 1: An example of the typical startup routine of TOS for a background data taking measurement at CAST for the Septemboard based GridPix detector. The indented lines refer to commands given to the previous command at top level.

pixel.

zero suppressed readout reads out only those pixels that have ToT values larger than 0, up to 4096 pixels. Stores the data in TSV files (tab separated values) 'X Y ToT' with an additional header. The header contains a global "run" and "event" header, which contains information about the run the event is taken from and a "chip" header, which contains information about the specific Timepix ASIC(s) being read out (up to 8 can be read out at the same time using TOS).

As for our purposes most events are extremely sparse (< 500 pixels active) the zero suppressed readout is the only relevant readout mode. The data files can be split into 3 distinct parts. A global run header, see listing 2, which contains information about the run the event is part of including important settings used as well as the timestamp of the event. Next is an event specific header, which contains specific information about the event in relation to the FADC and the scintillators, see listing 3. The final part of the zero suppressed data files is the chip header and tab separated value part of the 'X Y ToT' pairs of the active pixels for each chip of the detector in that event, see listing 4.

```
## [General]
## runNumber:        339
## runTime:          7200
## runTimeFrames:    0
## pathName:         data/runs/Run_339_190218-10-36
## dateTime:         2019-02-18.10:36:34
## numChips:         7
## shutterTime:      2
## shutterMode:      verylong
## runMode:          0
## fastClock:        0
## externalTrigger:  0
```

Listing 2: TOS generated data files start with a general header, which mainly contains information about the run the data file is part of. The only exception is the `dateTime` field, which represents the timestamp of the event (an oversight, should have been in the `[Event]` header).

FADC data files

If the FADC triggered during an event, as indicated by the `fadcReadout` field in the event header seen in listing 3, an additional data file is written with the same name as the event file, but a `.txt-fadc` extension. It contains a memory dump of the channels of the circular memory of the FADC plus a basic header about the FADC settings and the information about when the trigger happened.

The different fields in the header, see listing 5, are as follows:

```
## [Event]
## eventNumber:      2
## useHvFadc:        1
## fadcReadout:      1
## szint1ClockInt:   0
## szint2ClockInt:   0
## fadcTriggerClock: 647246
```

Listing 3: After the general header follows the event header in similar fashion. It records the event number and information about the FADC and scintillators. If the FADC triggered, `fadcReadout` is 1. Scintillator triggers may then be values in $[0, 4096)$. The `fadcTriggerClock` is the clock cycle of the Timepix frame in which the FADC trigger was received.

```
# chipNumber: 0
# chipName:   E 6 W69
# numHits:    0
# chipNumber: 1
# chipName:   K 6 W69
# numHits:    0
# chipNumber: 2
# chipName:   H 9 W69
# numHits:    2
106     160     75
211     142     2
```

Listing 4: The event header is followed by the beginning of the actual GridPix data. Each chip appears with a 3 line chip header containing number and name as well as the number of hits seen by that chip in the event. `numHits` lines follow with '`X Y TOT`' values in tab seperated fashion. This snippet would be followed by the remaining chips, as many as written in the run header 2 as `numChips`.

- `nb of channels`: decimal value of a 4-bit field that decides the number of active channels. `0` corresponds to using all channels as separate. We only use a single channel. [3]

- `channel mask`: decimal value of a 4-bit field to (de-)activate channels. `15` corresponds to all 4 channels active.

- `posttrig`: how many clock cycles in the 50 MHz [4] base clock of the FADC it continues taking data before commencing the readout (useful to record the rest of the signal and center it in the readout window)

- `pretrig`: the minimum acquisition time before a trigger is allowed to happen, in units of the 50 MHz [4] base clock.

- `triggerrecord`: together with `posttrig` allows to reconstruct the time of the trigger in the acquisition window

- `frequency`: decimal representation of a 6-bit field to select the operating frequency. `2 = 0b000010` corresponds to 1 GHz operation.

- `sampling mode`: decimal representation of a 3-bit field changing the operation mode (manual or automatic trigger) and register working mode (12 or 14-bit sensitivity of each register). We run in manual trigger and 12-bit mode. [3]

- `pedestal run`: a 1-bit flag indicating whether this file is a pedestal run.

The data portion starts with another semi-header of 12 data points, see listing 6. It contains fields that are not explained in the FADC manual, but instead refer to ”reserved for expert usage” [55]. One exception is data point 2, which is the so called Vernier, which could be used to determine the trigger time within two registers to get up to $\sim$ 50 ps RMS accurate time information. For our purposes though 1 ns time resolution is more than enough, given the signal undergoes integration and differentiation of a multiple of that in the shaping amplifier, anyway.

[3] As of writing this thesis, I don’t remember why the choice was made to only use a single channel instead of using all 4 channels to extend the time interval (development of these things happened between 2015-2017). It’s possible there were issues trying to combine all 4 channels. But it’s also just as likely it was an oversight due to lack of time combined with the fact that a 2.5 µs window is long enough for all intents and purposes. However, combining all 4 channels would even yield a long enough acquisition window when running in the 2 GHz sampling mode. Similarly, the choice of the 12-bit readout mode may represent plenty resolution in ADC values, but it seems prudent to not use the 14-bit mode given availability. All in all it leaves me head scratching (and thinking the likely reason will have been lack of time and being happy things working in the first place at the time).

[4] If running in 1 GHz mode. Else it corresponds to 100 MHz clocks.

```
# nb of channels: 0
# channel mask: 15
# posttrig: 80
# pretrig: 15000
# triggerrecord: 56
# frequency: 2
# sampling mode: 0
# pedestal run: 0
```

Listing 5: The file starts with a header indicated by #. Some of the values are decimal representation of bit fields, hence the weird values like ”0 channels”. It mixes both the configuration used as well as the time the trigger occurred (`triggerrecord`).

```
# Data:
# 3928
# 8022
# 3957
# 8076
# 3928
# 8023
# 3957
# 8077
# 2048
# 6138
# 2031
# 6151
```

Listing 6: After the header starts the data portion with some auxiliary information. The lines are neither of significant interest to us, nor are they properly explained in the manual. The second number corresponds to the Vernier, which can be used to determine the trigger more precisely than between individual register values, which is also not important for our purposes, as 1 ns resolution is plenty.

The final portion of the file contains the actual data (10 240 lines) and 3 fields at the very end to reconstruct the trigger within the acquisition window [5]. The data represents a pure memory dump of the cyclic register. See listing 7 for a shortened example.

```
2028 # register 0, channel 0
6119 # register 0, channel 1
1999 # register 0, channel 2
6100 # register 0, channel 3
2021 # register 1, channel 0
6108 # ...
... # 10240 lines of data in total
# 56
# 4096
# 0
```

Listing 7: Actual data portion of the FADC data. The first 10 240 lines represent a memory dump of the cyclic registers at trigger time (that is in their natural order instead of starting from the register in which the trigger was recorded). It starts at register 0 for channel 0, followed by register 0 of channel 1, and so on. As such each 4^{th} line corresponds to one channel. This is why the values jump so much from line to line. The last 3 lines are information to recover the trigger point in the acquisition window.

A.2.2. TOS configuration file

Everything related to the `HFManager` in TOS is controlled by a configuration file, normally located in `TOS/config/HFM_settings.ini`. The TOS configuration file used during CAST data taking is found in appendix B.1. We will go through the sections of it one by one and explain them.

Starting with the `[General]` section, listing 8. This section defines the VME related settings. The VME address of the HV module installed in the crate is used as the base address. The FADC address in the same VME crate is calculated from an offset in units of the VME address spacing of `0x0400`. [6]

The next section `[HvModule]`, listing 9, are general settings about the used HV module. The settings are related to the `KillEnable` feature of Wiener HV power supplies, the ramping speed of the HV channels and the time in seconds between sanity checks of the HV during data taking. [7]

[5] The last 3 lines of the data portion contain the trigger record, which is already printed by us in the header part and the `Valp_cp` and `Vali_cp` registers, which are only important if the FADC is used at a sampling frequency of $< 1\,\text{GHz}$, which is why we ignore it here.

[6] If in doubt about what the base address of the HV supply in the VME crate is, start one of the Wiener HV programs (for example `isegControl`), as it auto detects the module and prints the address.

[7] The `checkModuleTimeInterval` setting to check the HV status during the data taking was disabled at CAST, as it caused issues due to false alarms of the HV status. Given that the `KillEnable` flag was used, it was deemed unimportant. Attempting to fix it would have caused possible data loss as it would have been tested on the live detector.

```
[General]
sAddress_fadc = 1
baseAddress_hv = 0x4000
```

Listing 8: General section of the TOS configuration file. It sets the base address of the HV module installed in the VME crate. The FADC address is given as an offset from the base address.

```
[HvModule]
setKillEnable                   = true
# Voltage and Current RampSped currently set to arbitrary value
# in percent / second
moduleVoltageRampSpeed          = 0.1
moduleCurrentRampSpeed          = 50
# checkModuleTimeInterval       = 60, checks the status of the
# module every 60 seconds during a Run, between two events
checkModuleTimeInterval         = 60
```

Listing 9: The `[HvModule]` section contains settings related to the HV module as a whole. Whether a single channel tripping causes all channels to ramp down (`KillEnable`) the ramp speed and interval in which the HV module sanity status is checked.

Next up, the `[HvGroups]` section in listing 10 defines the different groups that combine multiple channels. There are multiple different kinds of groups in Wiener HV power supplies. The important groups are ramping groups and trip groups. Essentially, if one channel in a group starts ramping / trips all others also start ramping / shut off the HV, respectively. The section in the config file mainly exposes the already predefined sets of groups that are relevant for the Septemboard detector in TOS.

After the `[HvGroups]` section is the definition of the individual HV channels in listing 11. Here the physical channels on the device are mapped to the desired voltages and current bounds as well as to a human readable name. The fields repeat with increasing prefix numbers.

Second to last is the `[FADC]` section in listing 12. As the name implies it configures all parameters of the FADC. The main parameter to change is the `fadcTriggerThresholdRegisterAll` parameter, which defines the trigger threshold in effectively mV. Depending on the amount of noise in the system, adjustments to the threshold may be necessary.

The last section of the configuration file is the `[Temperature]` section, which deals with the safety ranges of the temperature of the detector. If the temperature leaves the safe range, the detector is to be shut down. [8]

[8] The temperature safety range is coupled to the `checkModuleTimeInterval` setting in the previous footnote. It was disabled together with the above during actual data taking.

```
# if this flag is set to true, anode and grid
# will be coupled to one group
[HvGroups]
anodeGridGroupFlag              = true
# grid is master channel of set on group
anodeGridGroupMasterChannel     = 4
anodeGridGroupNumber            = 0
monitorTripGroupFlag            = true
monitorTripGroupNumber          = 1
rampingGroupFlag                = true
rampingGroupNumber              = 2
gridChannelNumber               = 4
anodeChannelNumber              = 5
cathodeChannelNumber            = 8
```

Listing 10: `[HvGroups]` defines multiple groups of different HV channels together. The config file does not expose arbitrary groupings, but only sets flags whether groups are active and what their numbers are.

```
[HvChannels]
# all currents given in A (vmecontrol shows mA)
0_Name                          = grid
0_Number                        = 5
0_VoltageSet                    = 300
0_VoltageNominal                = 500
0_VoltageBound                  = 2.5
0_CurrentSet                    = 0.000050
0_CurrentNominal                = 0.000500
0_CurrentBound                  = 0
```

Listing 11: This is an excerpt of the full `[HvChannels]` section for a single HV channel. It maps the physical HV connectors to their voltages, current bounds and a human readable name. In this case the grid of the GridPixes of the Septemboard all receive a voltage of 300 V. The naming scheme of the fields is hardcoded for practical reasons and simply repeats with increasing numbers.

```
[Fadc] # FADC Settings
fadcTriggerType                 = 3
fadcFrequency                   = 2
fadcPosttrig                    = 80
fadcPretrig                     = 15000
# was 2033 before, 1966 corresponds to -40 mV
fadcTriggerThresholdRegisterAll = 1966
# run time of a single pedestal run for the FADC in ms
fadcPedestalRunTime             = 100
# number of acquisition runs done for each pedestal calibration
fadcPedestalNumRuns             = 10
# using channel 0 on FADC as trigger source, thus bit 0 = 1!
fadcChannelSource               = 1
# set FADC mode register (mainly to enable 14-bit readout)
fadcModeRegister                = 0b000
```

Listing 12: The [FADC] section configures the FADC. The most important setting is the trigger threshold as it defines the voltage required to trigger the FADC.

```
[Temperature] # temperature related parameters, all temps in °C
safeUpperTempIMB                = 61
safeUpperTempSeptem             = 61
safeLowerTempIMB                = 0
safeLowerTempSeptem             = 0
```

Listing 13: [Temperature] defines the safe operating ranges of the detector. If the range is left, the detector is to be shut down.

A.2.3. HV control via TOS

As is being alluded to in the previous section A.2.2, the HV control built into TOS can also handle ramping the channels of the detector. This is particularly convenient as it offers a very smooth ramping mode, which keeps the voltage potentials between all channels under a constant ratio. This allows for automatic ramping even for highly sensitive channels (like the GridPix grid).

In order to use the HV control and ramp the channels via TOS, `ActivateHFM` must be followed by `InitHV`, which attempts to connect to the HV power supply using the configuration of the config file. If the channels are not ramped up, a call to `RampChannels` will start the smooth ramping process (see listing 14).

```
1 > ActivateHFM
2 > InitHV
3 > RampChannels
```

Listing 14: The required commands to ramp the HV channels using the configuration from the config file using a smooth ramping mode.

If the HV is to be ramped down, a call to `ShutdownHFM` will ask whether the channels should be ramped down. There are a multitude of further commands available to communicate with the module, check the voltages, print status information etc.

Note that TOS can be started without ramping the HV channels and stopped without ramping the channels down. It is capable of connecting to a running HV power supply or leave it running after shut down.

A.2.4. Temperature monitoring

The two `MAX31685` micro controllers read out `PT1000` sensors, a group of **R**esistance **T**emperature **S**ensors (RTDs), which measure temperatures by its effect on the electrical resistance. They are platinum based and have a resistance of 1000 Ω at 0 °C. As the expected change in resistance is well understood, the temperature can be precisely measured.

The micro controller communicates with another micro controller, an `MCP2210` via the **S**erial **P**eripheral **I**nterface (SPI). SPI allows to address both `MAX31685` from the single `MCP2210`. The `MCP2210` is a USB-to-SPI micro controller. The USB connection from the intermediate board with the computer is separate from the rest of the detector communication. `TOS` communicates with it via the standard **H**uman **I**nteface **D**evice (HID) driver and utilizes an existing open source library for the `MCP2210` [231], which is slightly adapted.

The `ActivateHFM` command mentioned in the previous section also attempts to find the USB device

of the MCP2210 (the two are intertwined mainly, as the Septemboard detector is the only detector with either of the two features). If it is found, temperature logging starts immediately and the log files are placed in the default log directory of the TOS repository. Once a data taking run starts, the logging location is moved over to the data storage directory of the run. In either case the log file is named temp_log.txt and contains one temperature value for the intermediate board sensor (Temp_IMB) and one for the carrier board sensor (Temp_Septem) computed – based on an average over 5 s – and a timestamp (DateTime). A short snippet of the temperature log is shown in listing 15. [9]

```
# Temperature log file
# Temp_IMB  Temp_Septem   DateTime
26.5186     42.1472       2019-02-16.16:11:45
26.5217     42.2798       2019-02-16.16:11:51
26.5202     42.4371       2019-02-16.16:11:57
26.5309     42.5944       2019-02-16.16:12:03
26.5324     42.7347       2019-02-16.16:12:09
26.5355     42.8627       2019-02-16.16:12:15
26.5416     42.9707       2019-02-16.16:12:21
26.5432     43.0771       2019-02-16.16:12:27
26.5616     43.1804       2019-02-16.16:12:33
...
```

Listing 15: Snippet of a temperature log file as recorded for a run during the CAST detector lab measurements. Tabs were replaced by spaces for better visual alignment here.

[9] This default temperature logging location was also used as an unintended fallback mechanism during data taking, if the HV of the detector was considered out of certain bounds. Unfortunately, the bounds checking was entangled with the HV module sanity checks. As both features were very only implemented shortly before data taking, they triggered data taking aborts. For that reason the feature was disabled manually in code for the data taking at CAST. This however triggered a secondary code path for the temperature logging, storing it in the default location outside the specific run directories. As an effect the majority of CAST temperature logging data has been lost, as most of it was overwritten several times. Roughly daily manual temperature measurements are still left and show the detector operating in a good temperature range. Precise correlations with certain detector behaviors are unfortunately impossible. The two different code paths for the temperature logging are essentially a bug in the code that was never intended, stemming from the fact that temperature logging must be done to a 'global' location outside of data taking (as no data taking specific directory exists). Due to how the temperature logging and HV & FADC controls were added to TOS, these things were more entangled than necessary. A more thorough testing period of the detector and software package should have been performed, but was not in scope.

A.3. Septemboard event display

In order to monitor the data taking process while the detector is running, an online event display tool was developed during the first CAST data taking period in March 2017. It is a Python [10] based project making heavy use of `matplotlib` [114] for an interactive view of both the Septemboard events as they are recorded, the FADC readout as well as a general information header about the current data taking run.

The backend consists of a multiprocessing architecture with multiple worker processes. One process watches the current run directory for changes and reads the raw data files, another performs basic event reconstruction and a last one updates the current event to be displayed. The main process renders the `matplotlib` based graphical user interface (GUI) [11].

Fig. A.1 shows the graphical user interface of the septemboard event display during a background run. General information about the current run and event is shown in the box at the top center. The top left box shows hit specific information for the current event. The current septemboard event is always shown in the left pane in a realistic layout of the septemboard. By default the Viridis [12] color scale is used in the display of the septemboard events, each shown as an image of $(256, 256)$ pixels. If a chip did not record any activity during an event, its plot remains white for easier identification of few hits compared to no hits. The color scale can be adjusted when starting the program and the images can be downsampled by any factor of 2 for better visibility, as is done in fig. A.1. The right pane of the event display shows the last recorded event of the FADC. It does not automatically update the plot every time a new septemboard event is recorded, as there can be multiple events without FADC activity and it is useful to be able to glance at the last large event on the FADC. The filename is printed as the title of the plot to show which septemboard event it corresponds to.

The event display provides multiple forms of interactivity, such as an ”auto follow” mode (the default in which new events are shown as they are recorded), a ”playback” mode (walks through all events with a certain delay and general back and forth optionality. Further, a shortcut to save images directly exists, as well as simple computation of aggregate statistics of the current data taking period (different occupancy maps and simple histograms showing the number of hits per each event and chip).

All in all it provides a simple, but powerful way to monitor the detector activity online as it takes data. [13]

[10] `https://python.org`

[11] `matplotlib` provides a multitude of different GUI backends. The explicit choice depends on the specific machine (available backends may differ) and preference. Common choices are GTK and TkAg. `https://matplotlib.org/stable/users/explain/figure/backends.html`

[12] See here for the introduction of Viridis and its siblings: `https://bids.github.io/colormap/`

[13] The main drawbacks are related to it being a Python based project that utilizes `matplotlib` possibly too heavily. The combination means the tool is not useful for fast data taking, as it is too slow to show events in real time if data taking

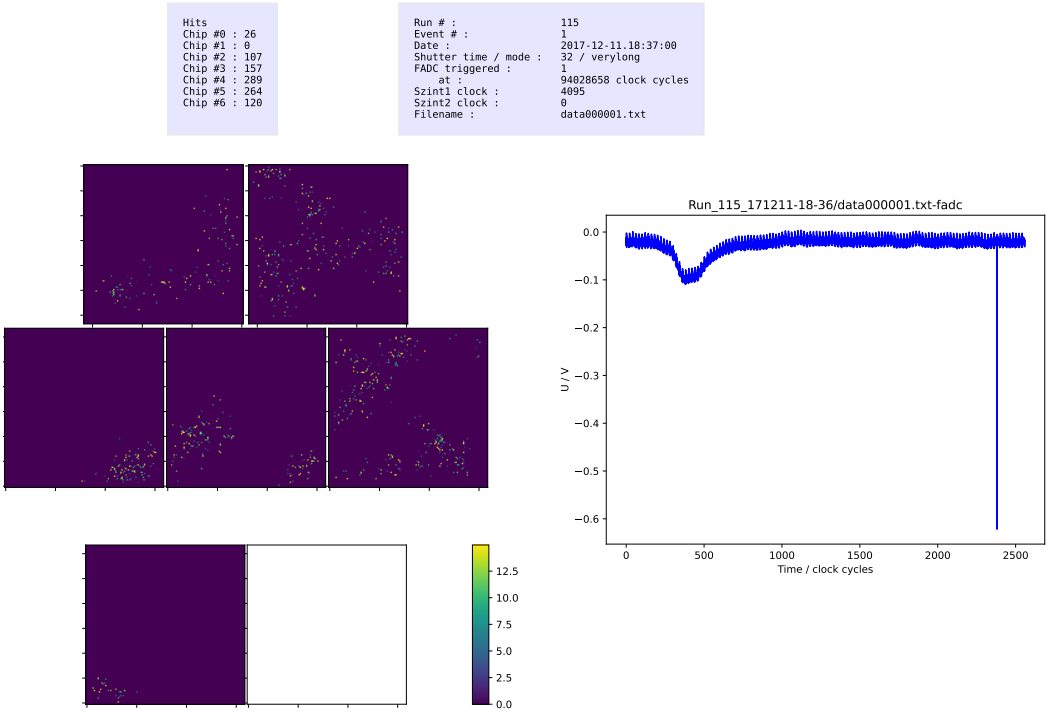


Figure A.1.: Screenshot of the Septemboard event display showing a background event from a CAST data taking run in 2017. The pixel density in the septemboard on the left has been downsampled by a factor of 2 from $(256, 256)$ for each chip to $(128, 128)$ for better visibilty of the activity. The event display shows general information like run and event number in the box at the top, hit specific information for the current event in the left top box and the current septemboard event in the left pane. The right pane shows the last FADC event (if no new FADC event is recorded, it stays).

exceeds one frame per second significantly.

B. Configuration and TOS / TOF versions Appendix

In this appendix we will present the TOS configuration file used for data taking at CAST and shortly note on versions of TOS and TOF used.

B.1. TOS configuration file

Below is the configuration file as it was used at CAST during the data taking periods. Most notably it contains the required address for the FADC in the VME crate, the high voltage settings (groups, voltages and current bounds) and the FADC settings (channel, trigger threshold etc.).

```
[General]
sAddress_fadc = 1
baseAddress_hv = 0x4000

[HvModule]
setKillEnable                   = true
# Voltage and Current RampSped currently set to arbitrary value
# in percent / second
moduleVoltageRampSpeed          = 0.1
moduleCurrentRampSpeed          = 50
# checkModuleTimeInterval       = 60, checks the status of the
# module every 60 seconds during a Run, between two events
checkModuleTimeInterval         = 60

# if this flag is set to true, anode and grid
# will be coupled to one group
[HvGroups]
anodeGridGroupFlag              = true
# grid is master channel of set on group
anodeGridGroupMasterChannel     = 5
anodeGridGroupNumber            = 0
monitorTripGroupFlag            = true
```

```
monitorTripGroupNumber          = 1
rampingGroupFlag                = true
rampingGroupNumber              = 2
gridChannelNumber               = 5
anodeChannelNumber              = 6
cathodeChannelNumber            = 9

[HvChannels]
# grid, anode and cathode settings
# all currents given in A (vmecontrol shows mA)
0_Name                          = grid
0_Number                        = 5
0_VoltageSet                    = 300
0_VoltageNominal                = 500
0_VoltageBound                  = 10
0_CurrentSet                    = 0.000050
0_CurrentNominal                = 0.000500
0_CurrentBound                  = 0

1_Name                          = anode
1_Number                        = 6
1_VoltageSet                    = 375
1_VoltageNominal                = 500
1_VoltageBound                  = 10
1_CurrentSet                    = 0.000050
1_CurrentNominal                = 0.000500
1_CurrentBound                  = 0

2_Name                          = cathode
2_Number                        = 9
2_VoltageSet                    = 1875
2_VoltageNominal                = 2500
2_VoltageBound                  = 15
2_CurrentSet                    = 0.000050
2_CurrentNominal                = 0.000500
2_CurrentBound                  = 0

3_Name                          = Ring1
3_Number                        = 7
3_VoltageSet                    = 415
3_VoltageNominal                = 500
3_VoltageBound                  = 15
3_CurrentSet                    = 0.000100
3_CurrentNominal                = 0.000500
```

```
3_CurrentBound                  = 0

4_Name                          = Ring29
4_Number                        = 8
4_VoltageSet                    = 1830
4_VoltageNominal                = 2500
4_VoltageBound                  = 15
4_CurrentSet                    = 0.000100
4_CurrentNominal                = 0.000500
4_CurrentBound                  = 0

6_Name                          = sipm
6_Number                        = 4
6_VoltageSet                    = 65.6
6_VoltageNominal                = 100
6_VoltageBound                  = 5
6_CurrentSet                    = 0.0005
6_CurrentNominal                = 0.0005
6_CurrentBound                  = 0

# The veto paddle scintillator is commented out, as it was supplied
# with HV by an external CAEN HV power supply.
# 5_Name                        = szintillator
# 5_Number                      = 11
# #5_VoltageSet                 = 1300
# 5_VoltageSet                  = 0
# 5_VoltageNominal              = 2500
# 5_VoltageBound                = 5
# 5_CurrentSet                  = 0.002
# 5_CurrentNominal              = 0.002
# 5_CurrentBound                = 0

[Fadc] # FADC Settings
fadcTriggerType                 = 3
fadcFrequency                   = 2
fadcPosttrig                    = 80
fadcPretrig                     = 15000
# was 2033 before, 1966 corresponds to -40 mV
fadcTriggerThresholdRegisterAll = 1966
# run time of a single pedestal run for the FADC in ms
fadcPedestalRunTime             = 100
# number of acquisition runs done for each pedestal calibration
fadcPedestalNumRuns             = 10
# using channel 0 on FADC as trigger source, thus bit 0  1!
```

```
fadcChannelSource               = 1
# set FADC mode register (mainly to enable 14-bit readout)
fadcModeRegister                = 0b000

[Temperature] # temperature related parameters
safeUpperTempIMB                = 61
safeUpperTempSeptem             = 61
safeLowerTempIMB                = 0
safeLowerTempSeptem             = 0
```

B.2. TOS and TOF versions used at CAST

For the DAQ software TOS unfortunately no discrete git tags exist, due to its rocky development. However, based on the git repository [215] and the dates of the start of the data taking campaigns (see 9.1), it is simple to deduce the corresponding commits which were used.

The detector firmware versions, TOF, have an even less well defined history. No development history exists, strictly speaking. The `.bit` files are available, but their names are very 'descriptive' in nature, but do not contain any version numbers either. If memory serves correctly, the version used in *Run-3* at CAST contained something like `szint1_fixed` in its filename, indicating that the scintillator trigger logic was fixed (which was the major bug in Run-2).

C. Calibrations Appendix

This appendix explains the different calibrations that need to be performed for the Timepix to bring it into its operating window, sec. C.1. Then in sec. C.2 we present additional calibrations for all chips.

C.1. Timepix calibrations

Before a Timepix based detector can be used for data taking, different calibrations have to be performed. We will discuss those calibrations, which are performed before any data taking here.

First, the THS optimization and threshold equalization (sec. C.1.1). These two are calibrations that are used to set different DACs on the Timepix to good working points. The very important ToT calibration was already introduced in the main body, section 7.1.1. Its purpose is to interpret the ToT values in amount of charge of recorded electrons. In principle there are many other calibrations one could perform, as the Timepix has 13 different DACs. Most are used with default values that are seen in tab. 7.3. S-Curve scans are introduced in sec. C.1.3, which can be used to map THL threshold DAC values to charges and determine the activation threshold. Similarly, the Pólya distribution can also act as a means to determine the activation threshold, see sec. C.1.4.

Important references for the Timepix in general and for the calibration procedures explained below are [142, 145, 143, 146].

C.1.1. THS optimization and threshold equalization

For an optimal operation of a Timepix based detector, each pixel should ideally have the same threshold. While all pixels are theoretically identical, imperfections in the production process will always lead to slight differences, either locally (transistor threshold voltage or current mismatches [142, 167]) or global effects like small supply voltage instabilities. Therefore, each pixel has 4 independently selectable current sources to minimize the spread of threshold values [143, 142]. Together all 4 sources act as an effective 4-bit DAC to slightly adjust the threshold. The absolute current for the 4 sources is dependent on the global THS DAC, allowing currents in the range of 0 to 40 nA.

To achieve a good calibration for a homogeneous threshold, first the THS DAC has to be set correctly. This is referred to as the THS optimization. Once the correct value is found, the 4-bit DAC on each pixel can be adjusted to minimize the spread of threshold values of all pixels together.

If the THS DAC is set too high, the 4-bit DAC on each pixel will be too coarse for a fine adjustment (as the 'current steps' will be too large). If it is too low, not enough range will be available to adjust each pixel to an equal noise / sensitivity level (not enough current available via the 4 current sources). The goal of the THS optimization is therefore to find just the right value, as to provide a range of values such that all values can be shifted to the same threshold of the threshold DAC THL.

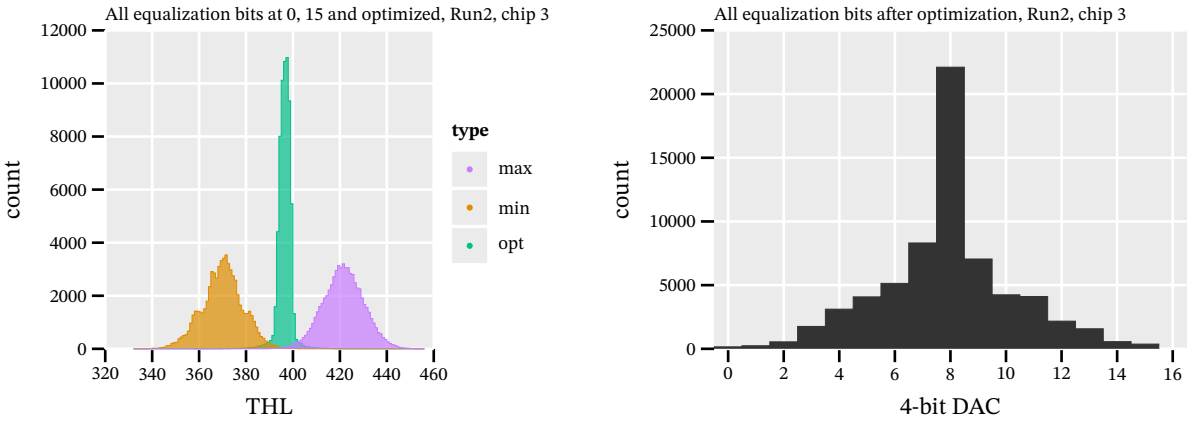
The algorithm scans a range of THL values through a subset of 4096 pixels using different 4-bit DAC values. First 0 for all pixels and then the maximum value of 15. At each THL and 4-bit value the number of hits due to pure noise is recorded for each pixel. The weighted mean of the THL values, using the number of hits as weight, is the value of interest for each pixel and each 4-bit DAC value: the effective THL noise value for that pixel. For each of the two cases (4-bit value 0 and 15) we can then compute a histogram of the number of pixels at each THL value. The resulting histogram will be a normal distribution around a specific THL value. The stopping criterion, which defines the final THS value, is such that these two distributions overlap at the 3 RMS level. This is performed by comparing the means of the 0 and 15 value distribution at a starting THS value and again at half of that THS value. Using linear regression of the two differences, the optimal THS value is computed.

With a suitable THS value set, the actual threshold equalization can start. The algorithm used is fundamentally very similar to the logic of the THS optimization. Each pixel of the chip is scanned for a range of THL values and the weighted THL noise mean is computed both at a 4-bit DAC value of 0 and at 15. The normalized deviation of each pixel's THL value to the mean THL value of all pixels is computed. Using a linear regression the optimal required shift (in units of the 4-bit DAC) yields the final 4-bit DAC value for each pixel.

An example of the 0 and 15 value distributions as well as the distribution using the final 4-bit DAC values for each pixel is shown in fig. C.1(a). [1] Each of the distributions represent different 4-bit DAC settings of all pixels of the chip. Orange ("min") represents all pixels using a 4-bit DAC value of 0, purple ("max") of 15. In green is the same distribution for the case where every pixel uses its optimal 4-bit DAC value. The threshold equalization thus yields a very strong reduction in the THL spread of all pixels. Fig. C.1(b) shows how all pixels are spread in the values of the 4-bit DAC. The narrow equalized line of fig. C.1(a) is achieved by a normal distribution around 8 of the 4-bit DAC values, with only very few at the edges of the DAC (0 and 15). Finally, fig. C.2 shows a heatmap of an entire chip with its 4-bit DAC values after equalization.

Similar plots for all other chips during both run periods can be found in the extended thesis.

[1] The plot is generated from the `thresholdMeans.txt` file created as part of the equalization procedure in TOS.



(a) THL distributions

(b) 4-bit distributions

Figure C.1.: (a)Distributions of different 4-bit DAC settings of all pixels of the chip. Orange (”min”): all pixels using a 4-bit DAC value of 0, purple (”max”): 15. Green: every pixel uses the optimal 4-bit DAC value after equalization. The result is a significant in =THL= value spread of all pixels.(b)Distribution of all 4-bit DAC values for the pixels after the threshold equalization. A normal distribution around a middle value is expected to largest likelihood of achieving a flat threshold around the whole chip. Very few pixels are either in value 0 or 15, implying few pixels likely outside their range to adjust to the required threshold. In this example represented is the center chip of the Septemboard with its calibration from July 2018.

C.1.2. Final THL (threshold) DAC value selection

Once the detector is THS optimized and threshold equalized, the final threshold value of the THL DAC can be determined for the data taking. While measurements like an S-Curve scan (see sec. C.1.3) can be used to understand where the noise level of the chip is in terms of THL values, it is typically not a reliable measure as the real noise depends strongly on the shutter length. If an experiment – like a low rate experiment as CAST – requires long shutter lengths, the best way to determine the lowest possible noise-free THL value is to perform a simple scan through all THL values using the shutter length in use for the experiment.

For a correctly equalized chip a sharp drop off of noisy pixels should be visible at a certain threshold. In principle the THL value at which no more pixels are noisy is the ideal THL value.

TOS first performs a quick scan in a THL range given by the user, using short shutter lengths. The determined drop values that still see some noise to a noise-free range is used as a basis for a long shutter length scan using a shutter length given by the user. For safe noise free operation one should choose a THL value 2 or 3 above the first noise-free THL value at the target shutter length. Especially for long shutter lengths it is important to perform this calibration without any high voltage applied to the detector as otherwise cosmic background starts to affect the data.

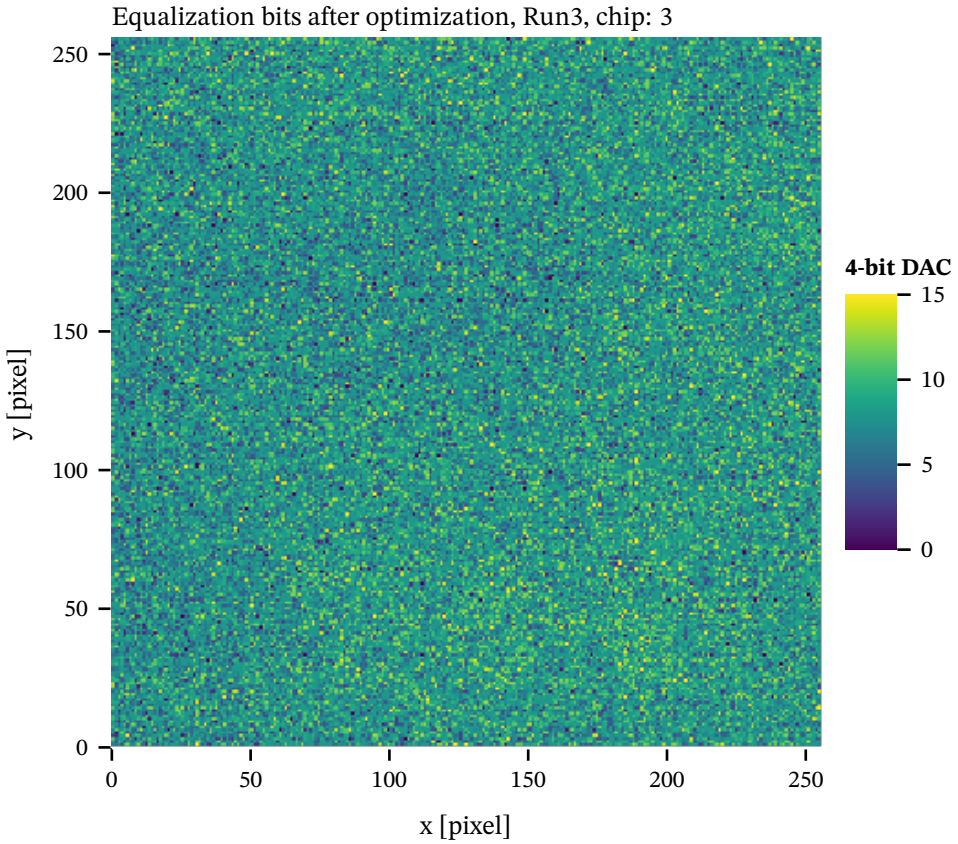


Figure C.2.: Heatmap of the distribution of the 4-bit DAC values of all pixels as they are spread over the full Timepix. In this example represented is the center chip of the Septemboard with its calibration from July 2018.

C.1.3. S-Curve scan

The S-Curve scan is one of 2 different ways to determine the optimal THL value.

The purpose of the S-curve scan is to understand the relationship between injected charges in electron and the THL DAC values by providing a # e^-/THL step number (or without a ToT calibration ToT/THL).

It works by injecting charges onto each pixel and checking the pixel response of each pixel at different THL values. Below a certain THL value all pixels will respond to the injected charge. At some point certain pixels will be insensitive to the induced charge and a 90° rotated "S" will form. By fitting an error function to this S an ideal THL value can be deduced.

By calculating THL value at which half of all test pulses are recorded, we can compute the number of electrons corresponding to that THL DAC value, as we know the amplitude of the test pulse and thus number of injected electrons.

Fig. C.3 shows an S-Curve scan of chip 0 of the Septemboard using the calibration from July 2018. The center peak in the middle is the noise peak of the detector at the shutter length used for the S-Curve scan. The symmetrical shape is due to specific implementation details of how the pixels function, the upper side is the one of interest. The center point (half way between both plateaus) corresponds to the effective threshold of the detector at that injected charge. The falling edge of each curve can be fit by the cumulative distribution function of a normal distribution, eq. C.1.

$$f(\mu, \sigma, N) = \frac{N}{2} \cdot \text{erfc}((x - \mu)/(\sigma \cdot \sqrt{2})) \tag{C.1}$$

where the parameter N is simply a scaling factor and μ represents the x value of the half-amplitude point. σ is the spread of the drop and erfc is the complementary error function $\text{erfc}(x) = 1 - \text{erf}(x)$. The error function is of course just the integral over a normal distribution up to the evaluation point x:

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} \, \mathrm{d}t.$$

Given that the number of injected electrons is known for each test pulse amplitude (see sec. 7.1.1), we can compute the relationship of the number of electrons per THL value step. This is called the THL calibration and an example corresponding to fig. C.3 is shown in fig. C.4, where the THL values used correspond to the μ parameters of eq. C.1. The resulting fit is useful, as it allows to easily convert a given THL DAC value into an effective number of electrons, which then corresponds to the effective threshold in electrons required to activate a pixel on average. When looking at the distribution of charges in a dataset, that cutoff in electrons is of interest (see sec. C.1.4).

Table tab. C.1 shows the fit parameters for the fits of fig. C.3. See the extended thesis for all fit parameters and plots.

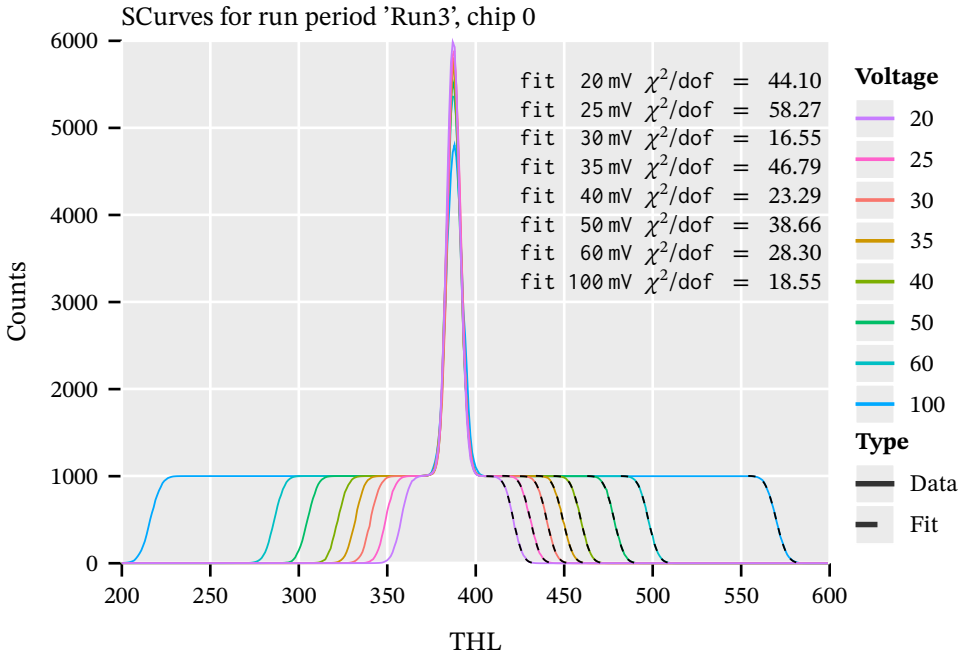


Figure C.3.: S-Curve scan of chip 0 of the Septemboard using the calibration from July 2018. The scan works by injecting 1000 test pulses at different amplitudes onto the pixels. Each line represents one such measurement and each point is the mean number of counted hits for all pixels with injected test pulses. The center peak in the middle is the noise peak of the detector. The symmetrical shape is due to specific implementation details of how the pixels function. The upper side is the one of interest. The falling edge of each curve can be fit by the complement of the cumulative distribution function of a normal distribution. The center point (half way between both plateaus) corresponds to the effective threshold of the detector at that injected charge. The fit parameters are found in tab. C.1.

V [U]	N	ΔN	μ	Δμ	σ	Δσ
20	1001	0.37	421.3	0.004754	4.221	0.006294
25	1004	0.3732	430.8	0.004994	4.545	0.006582
30	1000	0.3867	440.1	0.004989	4.429	0.006561
35	1005	0.3935	449.3	0.005154	4.627	0.006748
40	1002	0.3842	459.4	0.005048	4.53	0.006635
50	1002	0.3928	478.2	0.005145	4.603	0.006741
60	1001	0.3806	497.6	0.00505	4.553	0.006643
100	1004	0.3936	569.8	0.005365	4.895	0.007004

Table C.1.: Fit parameters of all S-Curves for Run-3 of chip 0 as shown in fig. C.3.

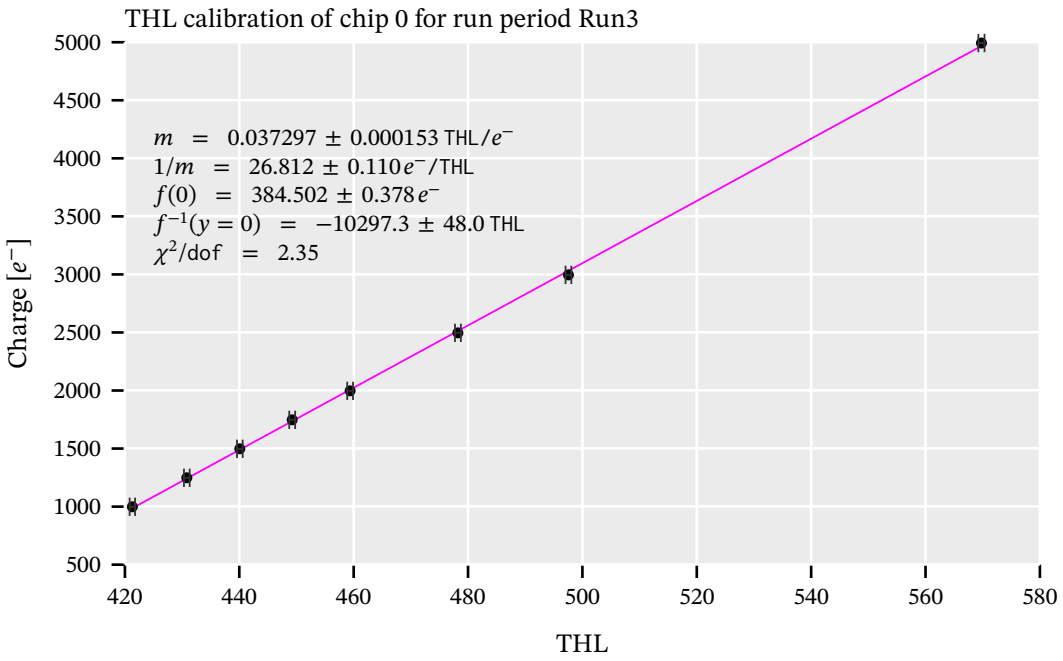


Figure C.4.: The THL calibration can be used to gauge the 'threshold gain' the THL DAC has on the number of electrons required to cross the threshold. The THL DAC in the Timepix normally adjusts the threshold by about 25 electrons per DAC value [143], which is reproduced well here (parameter $1/m$). The root of the linear fit (written as $f^{-1}(y = 0)$ in the annotation) corresponds to the position of the noise peak in fig. C.3.

C.1.4. Pólya distribution for threshold detection

In a gaseous detector the gas amplification (see sec. 5.3.6) allows to easily exceed the minimum detectable charge of $\mathcal{O}(500 \text{ to } 1000\,\mathrm{e}^-)$. The typically used THL threshold will be quite a bit higher than the 'theoretical limit' however for multiple reasons. One can either compute the effective threshold in use based on the THL calibration as explained in sec. C.1.3, or use an experimental approach by utilizing the Pólya distribution as introduced in sec. 5.3.6 and 7.1.2. By taking data over a certain period of time and computing a histogram of the charge values recorded by each pixel, a Pólya distribution naturally arises.

The Pólya distribution can be used to determine the actual activation threshold by simply checking what the lowest charge is that sees significant statistics.

An example of such a Pólya distribution with a very obvious cutoff at low charges is seen in fig. C.5. We see chip 0 using the same calibration from July 2018 as in the figures in the previous section C.1.3. The data is a 90 min interval of background data at CAST. The pink line represents the fit of the Pólya distribution to the data. The dashed part of the line was not used for the fit and is only an extension using the final fit parameters. The cutoff at the lower end due to the chip's threshold is clearly visible. The fit determines a gas gain of about 2700, compared to the mean of the data yielding about 2430.

Based on the data a threshold value of – very roughly – 1000 can be estimated. Using the THL calibration of the chip as shown in fig. C.4 yields a value of

$$Q(\text{THL} = 419) = 26.8 \cdot 419 - 10300 = 929.2$$

where we used the fit parameters as printed on the plot ($1/m$ and $f^{-1}(y = 0)$) and the THL DAC value of 419 as used during the data taking for this chip. Indeed, the real threshold is in the same range, but clearly a bit higher than the theoretical limit for this chip. This matches our expectation.

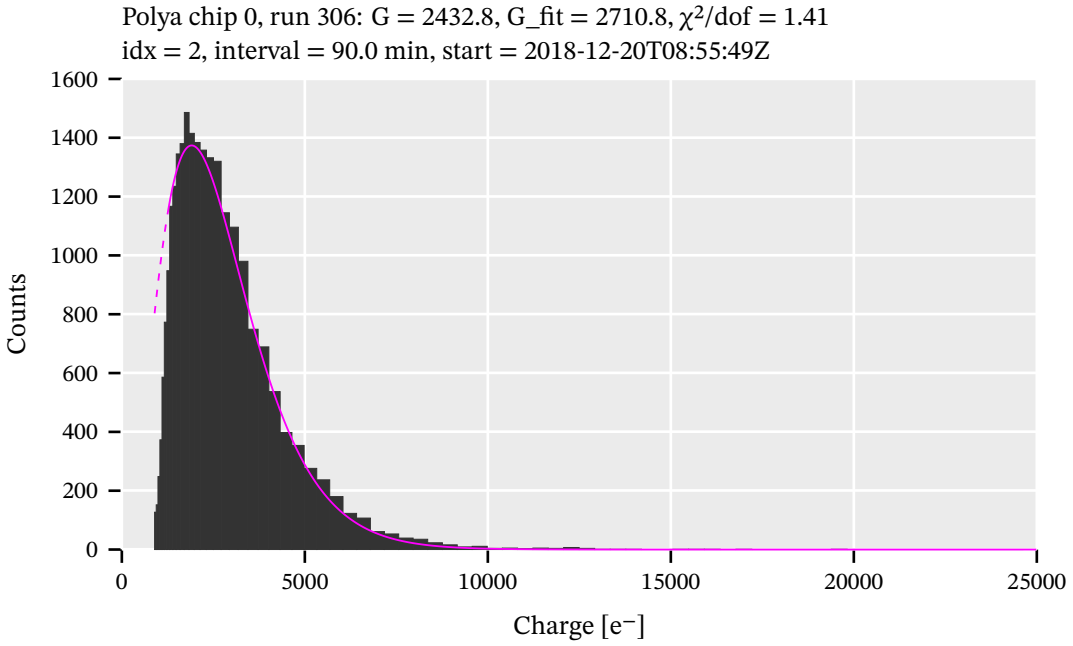


Figure C.5.: An example of a Pólya distribution of chip 0 using the calibration of July 2018 based on 90 min of background data. The lower cutoff is easily visible. The pink line represents the fit of the Pólya distribution to the data. In the dashed region the line was extended using the final fit parameters.

C.2. Septemboard calibration

In the extended thesis this section contains all THS optimization, S-Curves and ToT calibrations for both run periods and all chips of the Septemboard. As this leads to *a lot* of pages of figures, here we only show a histogram of all optimized THL distributions for each run period in sec. C.2.1.

C.2.1. THL calibration

Figure C.6 shows the optimized THL distributions of all chips after threshold equalization for run 2 (left) and 3 (right).

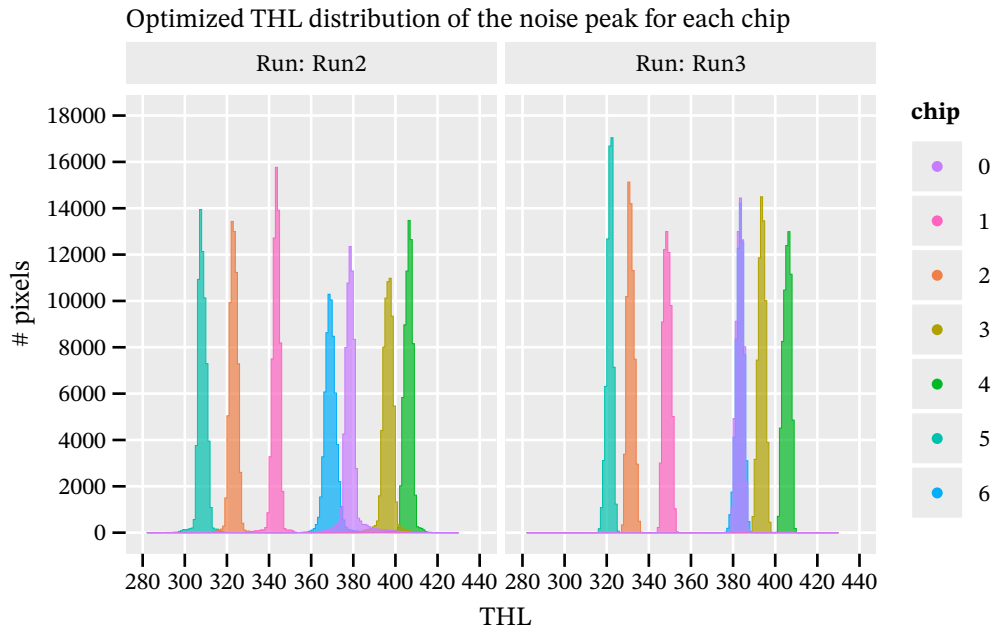


Figure C.6.: Distributions of the THL values of all Septemboard (board H) chips at the noise peak with the calibration for for run 2 on the left and run 3 on the right.

D. CAST operation procedures Appendix

This appendix provides some guidance about typical operations necessary during maintenance or installation of the detector, vacuum system and water cooling at CAST. The majority of this appendix was written as an overview of the Septemboard operation at CAST, as required by CERN at the time.

We first quickly look at the terminology to describe different areas in the CAST hall in sec. D.1. Then we go over the different systems. Detector high voltage (sec. D.2), vacuum system (sec. D.3), water cooling and gas supply (sec. D.4) and the safety interlock systems (sec. D.5).

We finally end with the CAST log files in sec. D.6. These are the most important aspect in terms of performing data analysis with the data taken at CAST (magnetic field, solar tracking and more).

D.1. CAST terminology

In the CAST collaboration there is a common terminology in use to describe different parts of the hall / different bores and detector installations. Essentially, the two magnet bores and areas 'behind' them (seen from the center of the magnet if you will) are named by their being closest to either the Geneva airport and by the Jura mountains on the other side.

The magnet ends are named by whether a detector installed on that side observes the sunrise or the sunset. See fig. D.1 for a schematic.

D.2. High voltage supply

The high voltage supply is an iseg HV module, which is located in the VME crate on the airport side of the magnet. The HV is controlled via a USB connection to the VME crate, which it shares with the FADC. The veto scintillator however has its own HV supply, since it needs a positive HV, instead of a negative one.

The detector uses 7 different high voltages. 5 of these are for the detector itself, 1 for the SiPM and the last for the veto scintillator on top. Their voltages are shown in tab. D.1. The voltages actually used are defined in TOS/config/$\text{HFM}_{\text{settings.ini}}$.

The HV cables in use are red cables with LEMO HV connectors. They run from the detector to an iseg HV module sitting in a VME crate on the airport side of the magnet. The cables are marked with

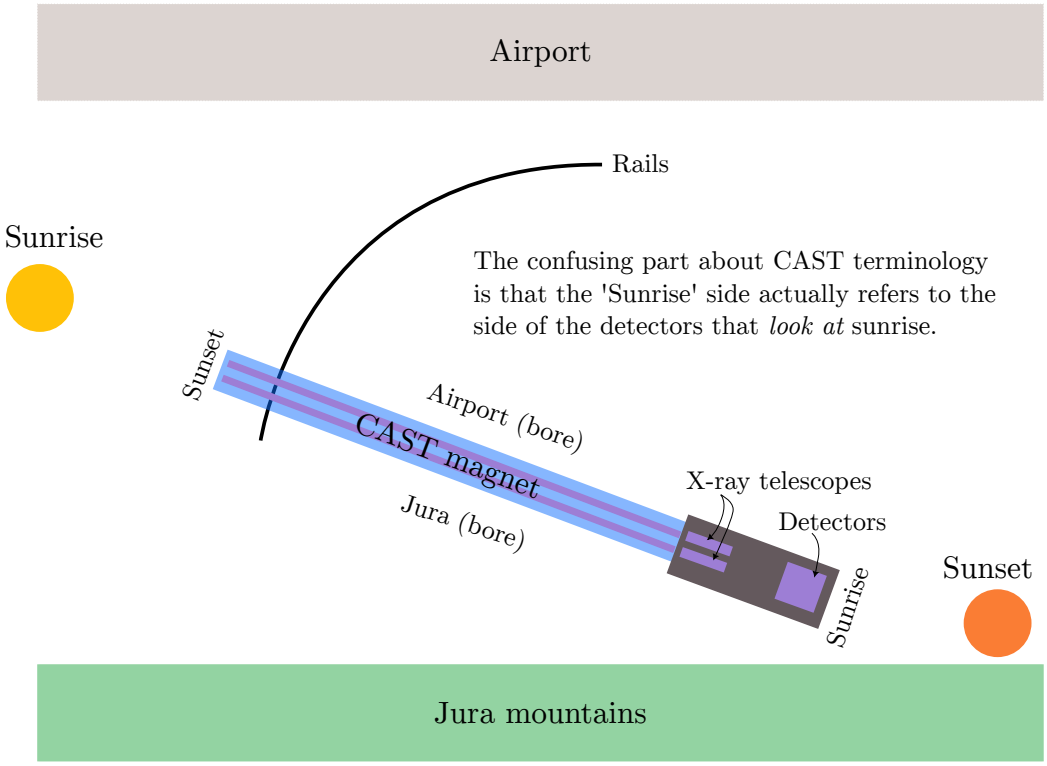


Figure D.1.: Schematic of the CAST hall with the magnet and the parts that give the names to different areas. The bores (and as an extension sides of the magnet) are named by the airport / Jura mountains and the magnet ends by whether a detector observes sunrise or sunset.

Description	Channel	Voltage / V	TripCurrent / mA
grid	0	-300	0.050
anode	1	-375	0.050
cathode	2	-1875	0.050
ring 1	3	-415	0.100
ring 29	4	-1830	0.100
veto scinti	5	+1200	2
SiPM	6	-65.6	0.05

Table D.1.: Table of high voltages in use for the Septemboard detector. Note that the veto scintillator is not controlled via the iseg module, but by a CAEN N470.

zip ties and the same names as in tab. D.1.

The interlock system for the high voltage supply is detailed in section D.5.1, together with the other interlock systems in place.

D.2.1. Ramping the HV

The high voltage supply can be controlled in two different ways. Besides differing in usability terms (one is manual, the other automatic), the main difference between the two is the HV interlock, which is only partially usable in case of the manual HV control.

1. in the manual way using the Linux software supplied by iseg. On the InGrid-DAQ computer it is located in ~/src/isegControl/isegControl. Depending on the setup of the machine, the software may need superuser rights to access the USB connection. With the software the given channel as shown in tab. D.1 can be set up and the HV can be ramped up. Note: one needs to activate SetKillEnable such that the HV is shut down in case of a current trip (exceeding of specified current). One should then set 'groups' of different channels, so that grid, anode and ring 1 are shut down at the same time, in case of a current trip, as well as ring 29 and the cathode! In addition the trip current needs to be manually set about a factor 5 higher during ramping, because of capacitors, which need to be charged first. Otherwise the channels trip immediately. In this case the HV interlock is restricted to basic current restrictions. Anything detector related is not included!

2. in the automatic way via the TOS. The TOS takes care of everything mentioned above. To use the TOS for the HV control (and thus also use the complete HV interlock, as it exists at the moment), perform the following steps:

   a) check ~/TOS/config/$\text{HFM}_{\text{settings.ini}}$ and compare with tab. D.1 whether settings seem reasonable

   b) after starting TOS and setting up the chips, call

```
1       > ActivateHFM
```

   which will set up TOS to use the combined HV and FADC (due to both inside the same VME crate, they are intertwined). This configures the FADC and reads the desired HV settings, but does not set the HV settings on the module yet.

   c) to write the HV to the HV module, call

```
1       > InitHV
```

which will write the HV settings from `HFM_settings.ini` to the HV module. At the end it will ask the user, whether the HV should be ramped up:

```
1       Do you wish to ramp up the channels now? (Y / n)
```

If yes, the ramping progress will be shown via calls to the CheckModuleIsRamping() function (which can also be called manually in TOS).

This should properly ramp up all channels. It is possible that TOS fails to connect to the VME crate and hence is not able to ramp up the channels. The most likely reason for this is that the isegControl software is still open, since only one application can access a single USB interface at the same time.

D.3. Vacuum system

This section covers the vacuum system of the detector. It is pumped via a single primary membrane pump and two turbo pumps. One turbo pump is used to pump the main vacuum vessel of the beam pipe, while the second small turbo pump is used to pump the interstage part of our X-ray source manipulator to reduce leakage during movement of the source.

Fig. D.2 shows a schematic of the whole vacuum system including all interlock systems and the pressure sensors. The pressures of the sensors P3 and P-MM are used as an interlock for VT3 and the gas supply, respectively.

D.3.1. Vacuum operations

The vacuum system as described in sec. D.3 usually does not require manual intervention during normal operation.

For maintenance the following two sections describe how to pump the system safely as well as how to flush it with nitrogen. Both processes are rather delicate due to the sensitive $Si_x N_y$ window. Pumping needs to be done slowly, $O(1\,\mathrm{mbar\,s^{-1}})$. To be able to do this, the needle valve V_{Needle} (cf. fig. D.2) is installed. One may separate the vacuum volume into two separate vacua. A bad vacuum before the primary pump and after the turbo pumps T1 and T2 and a good vacuum before the turbo pump T2. There are three connections from the good vacuum to the bad one.

1. through T2, closable via V_{T2}, 40 mm tubing
2. through the needle valve V_{needle}, 16 mm tubing
3. through T1 via the manipulator interstage, normally closed (see the note below), 25 mm tubing

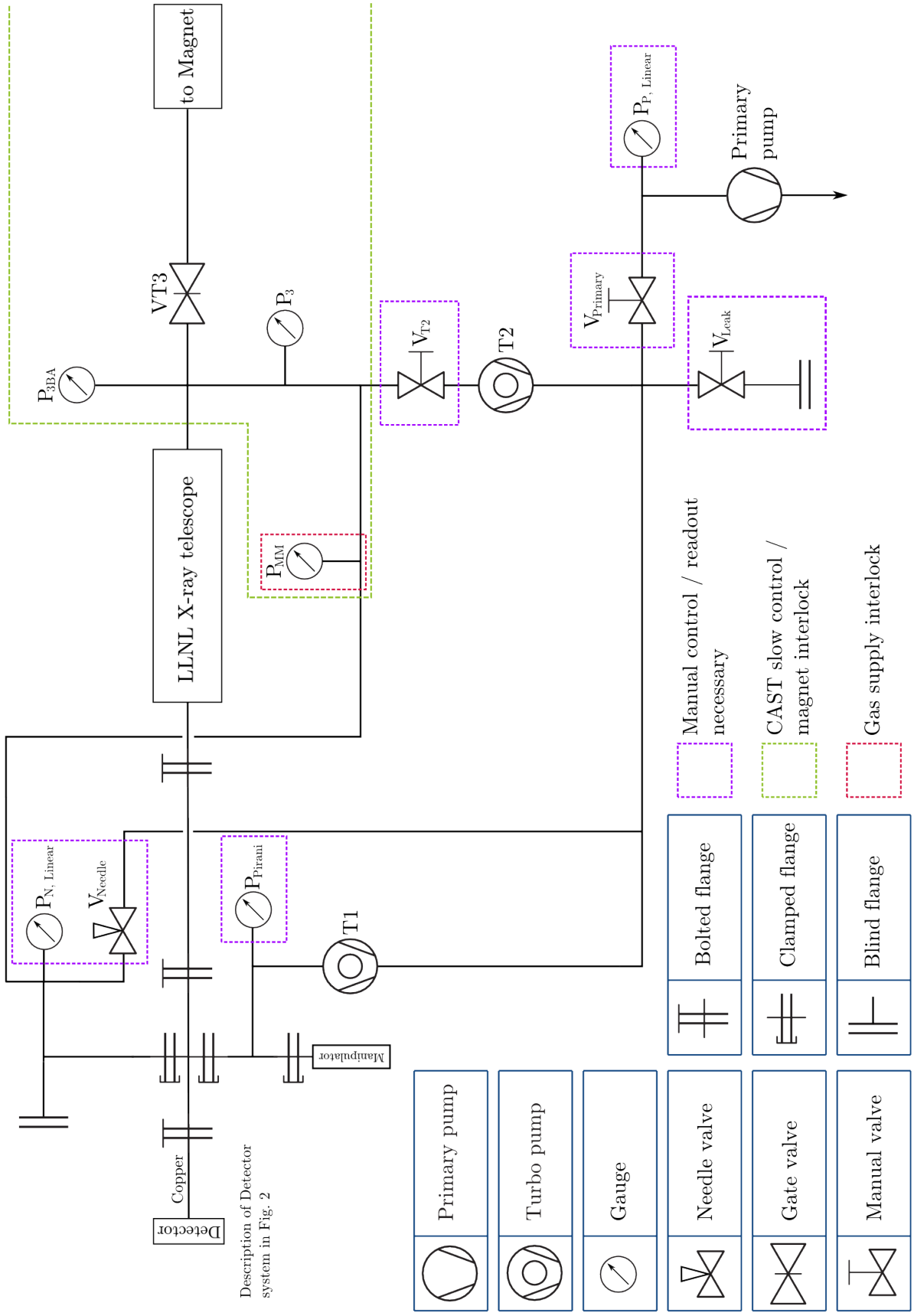


Figure D.2.: Schematic of the vacuum system behind the LLNL telescope including interlocks and pressure sensors.

3 is mainly irrelevant for pumping purposes, since there is no valve to open or close; it is always closed by a 2-O-ring seal to the good vacuum. While 1 is the main path for pumping during operation, it is 2 which is used during a pumping down or flushing of the system, since it can be controlled very granularly.

For both explanations below, it is very important to always think about each step (are the correct valves open / closed? etc.). A small mistake can lead to severe damage of the hardware (turbo pumps can break, the window can rupture).

Note: There is a third very small vacuum volume before T1, which is the volume up to the manipulator interstage. This volume is separate from the main good vacuum chamber, due to a 2-O-ring seal on both ends of the manipulator. Compare with fig. D.2 at the location of the two clamped flanges 'above' the manipulator. One 2-O-ring seal is at the upper flange and one at the lower. This is because the manipulator part furthest from the beampipe is under air. In order to seal the air and the vacuum especially during movement of the source, these seals are in place. However, while the 2-O-ring seals provide decent sealing, it is not perfect. This is why the small turbo pump T1 is in place at all, to reduce the amount of air, which might enter the system during source manipulation. Another aspect to keep in mind is potential air, which can get trapped in between the two O-rings. This air will be released during movement of the seals. Especially after the system was open to air, it is expected that a small pressure increase on P_{MM} can be seen during operation, despite T1 being in place. After several movement cycles, $O(10)$, these peaks should be negligible.

Pumping the vacuum

Before pumping it is a good idea to connect two linear gauges to the two P_{Linear} pressure sensors. To pump the system safely, perform the following steps:

1. Make sure every pump is turned off.
2. Make sure every valve in the system is closed:
   a) V_{Primary}
   b) V_{Leak}
   c) V_{T2}
   d) V_{Needle}
3. Connect a linear gauge to $P_{\text{P, Linear}}$ on the primary pump line.
4. Start the primary pump. Tubing up to V_{Primary} will be pumped, visible on linear gauge connected to $P_{\text{P, Linear}}$. Check that the second linear gauge remains unchanged, if not V_{Primary} and V_{Needle} is open!

5. Once $P_{\text{P, Linear}}$ shows $\leq$ 10 mbar, slowly open V_{Primary}, again checking $P_{\text{N, Linear}}$ remains unchanged. This will increase the pressure on $P_{\text{P, Linear}}$ again until the volume is pumped.
6. This step is the most crucial. With V_{T2} still closed, very carefully open V_{Needle}, while keeping an eye on $P_{\text{N, Linear}}$. Note that V_{Needle} has two locking mechanisms. The knob at the end with the analog indicator and a general lock in front of that. While the analog indicator shows `000`, open the general lock. Then slowly start turning the knob. At around `300` the pressure on $P_{\text{N, Linear}}$ should slowly start to decrease. Keep turning the knob until you reach a pump rate of $O(1\,\text{mbar}\,\text{s}^{-1})$. You will have to keep opening the needle valve further, the lower the pressure is to keep the pump rate constant.
7. Once both linear gauges have equalized (up to different offsets), close the needle valve again.
8. Open V_{T2}.
9. Start T2 by turning on the power and pressing the right most button. Use the arrow buttons to select the 'actual RPM' setting to see that the turbo is spinning up. Final speed should be set to 1500 Hz.
10. While T2 is spinning up, start T1 by turning on the power at the back. There is no additional button to be pressed.

The system should now be in the following state:

- V_{Leak} closed
- V_{Needle} closed
- V_{T2} open
- V_{Primary} open
- T2 & T1 running
- Primary pump running

If so, the system is now pumping. Note that it may take several days to reach a vacuum good enough to satisfy the interlock.

Flushing the system

Flushing the system is somewhat of a reverse of pumping the system. Follow these steps to safely flush the system with nitrogen. See section D.4.4 for an explanation of which valves need to be operated to open the nitrogen line. Before flushing the system connect two linear gauges to both P_{Linear} sensors.

1. Make sure the turbo pumps are turned off, if not yet turn both off and wait for them to have come to a halt.

2. Turn off the primary pump.

3. Close V_{T2}. V_{Leak} and V_{Needle} should already be closed, while V_{T2} and V_{Primary} should still be open.

4. Connect the nitrogen line to the blind flange before V_{Leak}.

5. Slowly open V_{Leak}, while checking both linear gauges. Make sure only the pressure on $P_{\text{P, Linear}}$ increases, while $P_{\text{N, Linear}}$ remains under vacuum. If not, another valve is still open. Close V_{Leak} immediately again!

6. Keep flushing nitrogen, until $P_{\text{P, Linear}}$ gets close to 1000 mbar (the sensors will never actually reach that value).

7. Close V_{Leak} again to make sure you do not put the system over one atmosphere of pressure.

8. This step is the most crucial. With V_{T2} still closed, very carefully open V_{Needle}, while keeping an eye on $P_{\text{N, Linear}}$. Note that V_{Needle} has two locking mechanisms. The knob at the end with the analog indicator and a general lock in front of that. While the analog indicator shows `000`, open the general lock. Then slowly start turning the knob. At around `300` the pressure on $P_{\text{N, Linear}}$ should slowly start to increase. Keep turning the knob until you reach a pump rate of $O(1\,\text{mbar}\,\text{s}^{-1})$.

9. You will notice that the pressure on $P_{\text{P, Linear}}$ will start to decrease, since the air will distribute in a larger volume. Open V_{Leak} again slightly to keep $P_{\text{P, Linear}}$ roughly constant.

10. Keep flushing with $1\,\text{mbar}\,\text{s}^{-1}$ until both sensors read $O(1000\,\text{mbar})$.

11. Close all valves in the system again.

This way the system is safely flushed with nitrogen. This helps to pump faster after a short maintenance, because less humidity can enter the system.

D.4. Watercooling system & gas supply

In this section the watercooling system as well as the gas supply is discussed. In section D.4.3 a combined schematic of both systems is shown.

D.4.1. Watercooling

In order to keep the detector cool enough to avoid noise and damage to the septemboard, a watercooling system is used. This section describes the relevant information for the system as used at CAST.

To readout the temperature two PT1000 temperature sensors are installed on the detector. One is located on the bottomside of the intermediate board (outside of the detector volume), while the other is located on the bottom side of the Septemboard. This temperature T_{Septem} is also included in the schematic D.6 below, because it was intended to be part of the HV interlock, as described in D.5.1. In the end it was not due to lack of time for proper testing. [1]

Fig. D.3(b) shows the main part of the system including the pump, reservoir and radiator. The tubing is specifically chosen in **blue** to clear up potential confusion with other tubes used in the detector system. [2] These tubes use special Festo quick couplings, which cannot be connected to the connectors of the gas supply, to avoid potential accidents. The tubes have zipties installed on them, which label the tubes as well, with the naming convention as it is used in fig. D.6.

Maintenance At the end of every shift it should be checked, whether the water level in the reservoir is still above the red line seen in Fig. D.3(a). If not, water should be added by the shift coordinator (or trusted shifters).

D.4.2. Gas supply

The gas supply uses red tubing (in parts where flexible tubing is used) to differentiate itself from the watercooling system. Additionally, the tubes have zipties showing which tube is which. These are located on both ends of the tubes. The naming convention is the same as in D.6. The connectors of the gas line are standard Swagelok connectors.

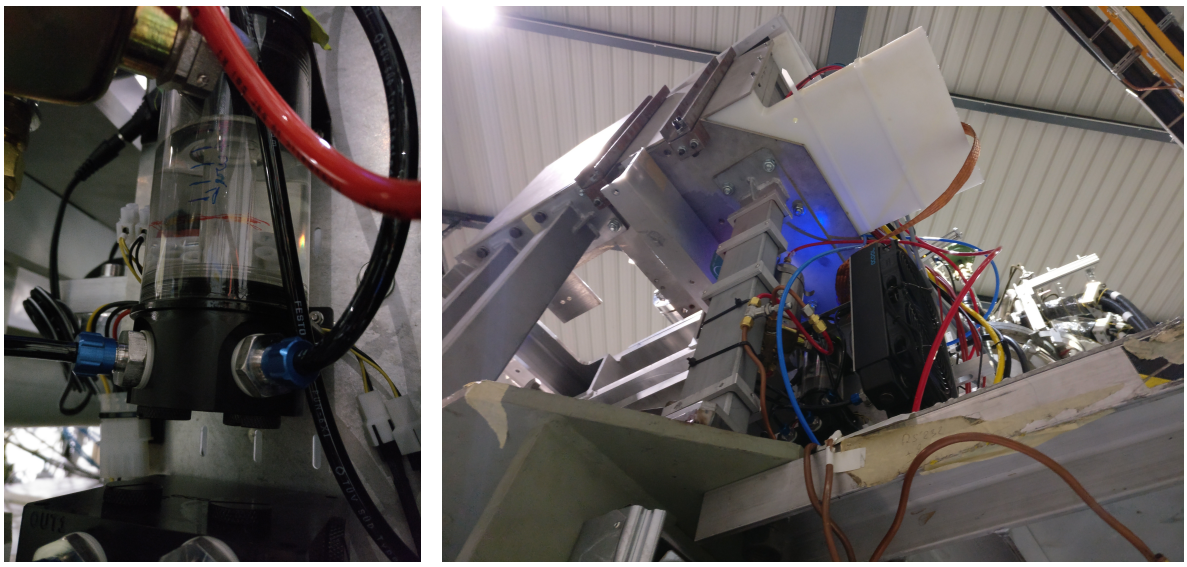
As can be seen in the schematic, the gas supply has 4 valves on the inlet side and 2 on the outlet side. In addition a buffer gas volume is installed before the detector for better flow control.

It follows a short explanation of the different valves:

- $V_{\text{in, 1}}$ is the main electrovalve installed right after the gas bottle outside the CAST hall.
- $V_{\text{interlock}}$ is installed is the electrovalve installed below the platform where the beamline is located. This valve is part of the gas supply interlock, as described in section D.5.2.
- $V_{\text{in, 2}}$ is the manual valve located on the second gas supply mounting below the beamline.

[1] As mentioned in a previous footnote, the decision to take it out of the HV interlock is the reason for loss of temperature logging data.

[2] This change was made after the 'window accident'.



(a) Reservior (b) Water cooling system

Figure D.3.: (a)shows a closeup of the water cooling reservoir with indicated markings to be checked for safe water levels.(b)is a look from the ground floor at the water cooling system including radiator, pump and reservoir.

- $V_{\text{in, N1}}$ is the first needle valve, which is located on the second gas supply mounting below the beamline. It is the one part of the flow meter placed there.
- V_{P} is the valve inside the pressure controller, which is placed roughly below the telescope (on the platform), while P_{Detector} is the pressure gauge inside this controller.

Gas system operations

If the requirements of the gas supply interlock are satisfied (cf. sec. D.5.2), it is possible to flush the detector with gas. For that, follow these steps:

1. Make sure the pressure controller is connected and running. Check InGrid-PLC computer in control room and see if pressure control software is running. If gas supply is currently closed, reported pressure inside the detector is usually reported to 960 to 980 mbar.
2. Outside the building, open the main valve of the currently active gas bottle (check the arrow on the bottle selector mechanism). See fig. D.4(a).
3. Open the second valve near the bottle.
4. Pressure values should be:
    - gas bottle: $\sim$ 30 to 100 bar

- pre-line pressure: $\sim 7\,\mathrm{bar}$
- line pressure: $\sim 0.45\,\mathrm{bar}$

5. Activate the gas supply at the interlock box by turning the key to `Security on` and pressing the large button. See fig. D.4(b).

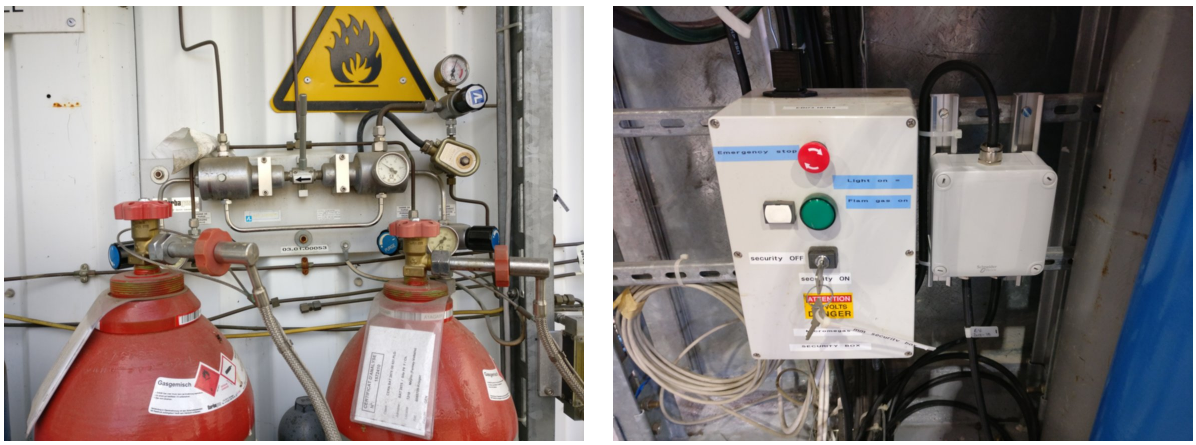
6. Go to the airport side of the magnet. Open the valve on the InGrid gas panel below the telescope platform. See fig. D.5.

7. Slowly open the needle valve on the flow meter on the previous panel. Increase gas flow up to $\sim 2\,\mathrm{L\,h^{-1}}$.

8. Open the needle valve on the gas supply line on the side of the platform.

9. After 5 to 10 min the pressure controller on the InGrid-PLC computer should report 1050 mbar.

Now the detector should be flushed with $\mathrm{Ar/iC_4H_{10}}$. Before turning on the HV, make sure to flush for at least 12 h to be on the safe side.



(a) Gas bottle (b) Interlock box

Figure D.4.: (a): Location of the argon-isobutane bottle and the main valves outside the building.(a): Location of the gas interlock box

D.4.3. Combined schematic (water & gas)

Fig. D.6 shows a combined schematic of both the watercooling system and the gas supply. Additionally, the relevant interlock systems and their corresponding members are shown.

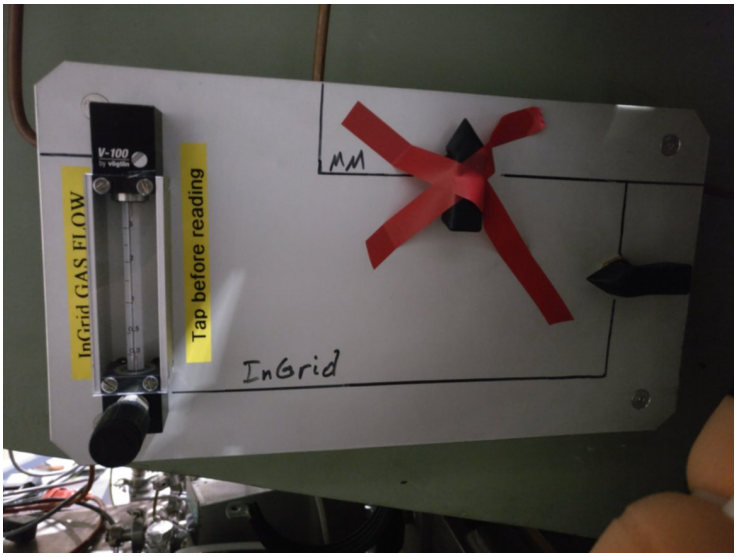


Figure D.5.: Location of the InGrid gas panel below the telescope platform on the airport side.

D.4.4. Nitrogen supply

Nitrogen is supplied by a nitrogen bottle outside the building. To open the nitrogen line, 5 valves need to be opened. The line ends in a copper pipe on the airport side, which is usually there rolled up (the copper is somewhat flexible).

1. Open the lever on the nitrogen bottle outside the building (see fig. D.7(a)).
2. Open the valve next to the bottle.
3. Go to the gas lines next to the control room. The right most line (see fig. D.7(b) and fig. D.8(a)) is the nitrogen line. Open the valve.
4. Open the needle valve on the flow meter to the right of the previous valve.
5. Open the needle valve on the airport side of the magnet.

This should be all to open the nitrogen line. The flow through the pipe is not too large, but it should be large enough to feel it on the back of the hand.

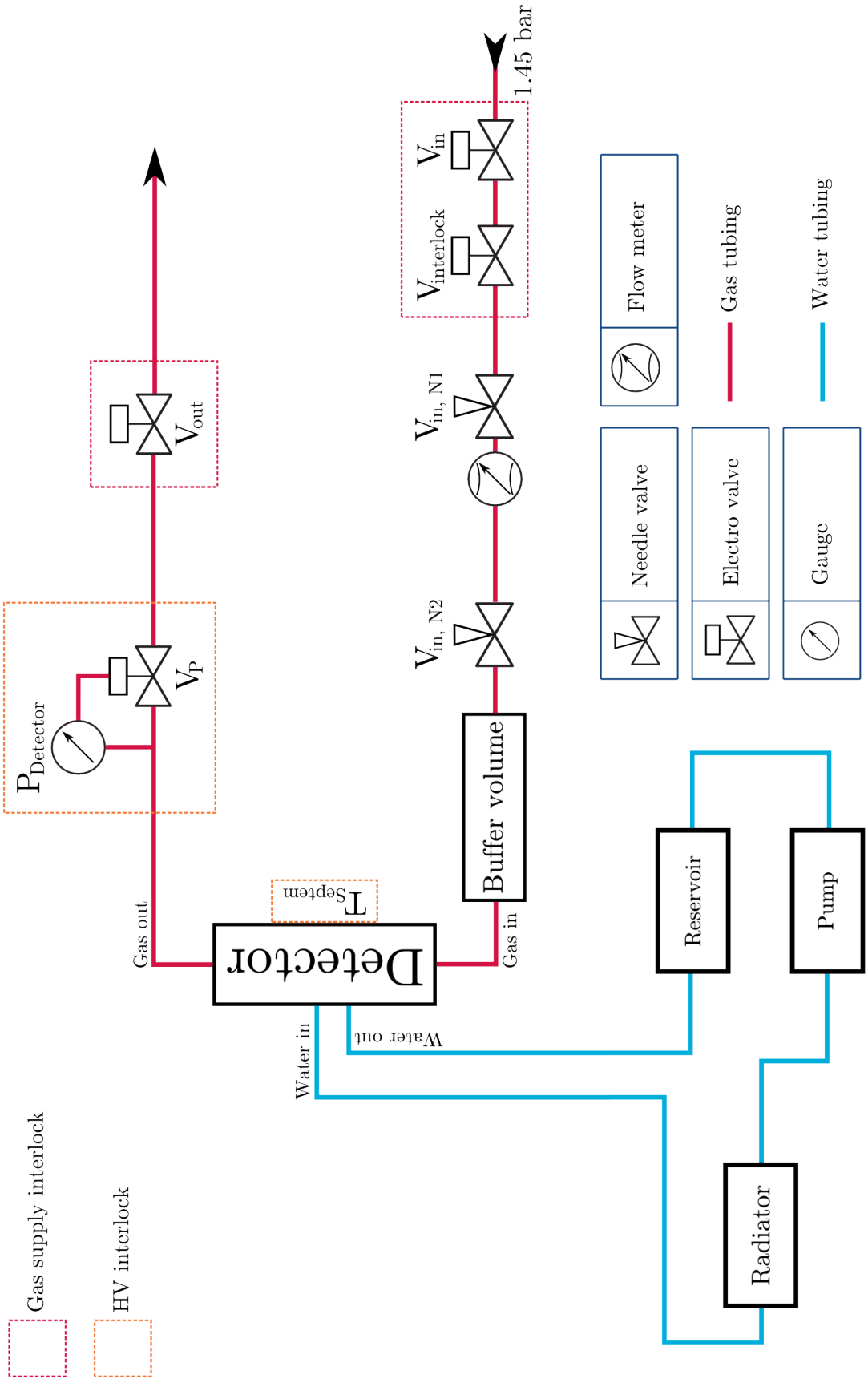
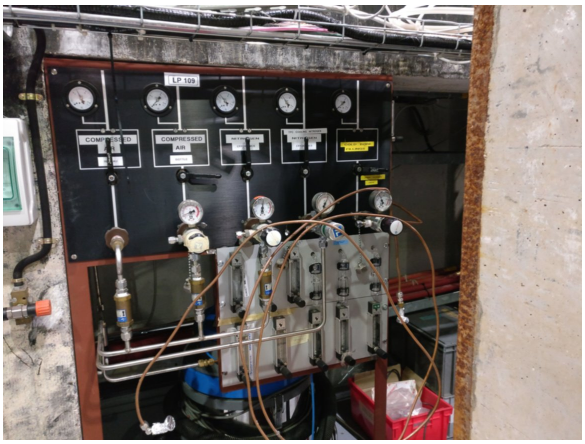


Figure D.6.: Combined schematic of the detector system, consisting of the water cooling system and the gas supply. The interlock systems are shown with dashed lines. See section D.5 regarding explanations of when the interlock is activated.

(a) Nitrogen bottle



(b) Control room

Figure D.7.: (a): Location of the Nitrogen bottle and the main valves outside the building.(b): Location of the valves next to the control room

(a) Nitrogen valve

(b)

Figure D.8.: (a): The actual nitrogen valve on the set of valves near the control room.(b): Location of the flow meter near the valve next to the control room

D.5. Interlock systems

This section describes the interlock systems, which are related to the detector. There are 3 interlock systems to speak of.

- The CAST magnet interlock, which prohibits the gate valve VT3 to be opened, if the pressure in the vacuum system is not good enough.
- A gas supply interlock, which makes sure the detector is only flushed with gas, if all other parameters are considered nominal (mainly a good vacuum in the system).
- A HV interlock, which makes sure the detector is only under HV, if the temperature of the detector is still good (otherwise a lot of sparks are produced) and the currents on the HV lines are nominal.

D.5.1. HV interlock

> Note: This section describes the high voltage interlock as it was intended. But as mentioned multiple times previously it was deactivated for final data taking due to several bugs causing it do trigger under non intended circumstances.

The HV system is part of an interlock, which tracks the following properties:

- detector temperature
- currents on HV lines
- TO BE IMPLEMENTED: gas pressure inside the detector [3]

The detector temperature is measured at two points by PT1000 sensors. One of these is located on the bottom side of the intermediate board (and thus is more a measure the the temperature surrounding the detector), while the second is located on the bottom side of the septemboard. The second is the best possible measure for the temperature of the InGrids. However, there is still a PCB separating the sensor from the actual InGrids. This means there is probably a temperature difference of a minimum of 10 °C between the measured value and the actual temperature of the InGrids.

Whenever the TOS is configured to use the FADC readout and control the HV (note: the two are intertwined, since both sit in the same VME crate, which is controlled via a single USB connection), a background process which monitors the temperature is started. If the temperature exceeds the following boundaries

[3] The gas pressure inside the detector was intended to be added to the HV interlock as well, but also due to lack of time it never was.

$$0\,°\text{C} \leq T \leq 60\,°\text{C}$$

on the lower side of the septemboard, the HV is shut down immediately. The lower bound is of less practical value in a physical sense, but in case of sensor problems negative temperature values may be reported. As the upper bound a value is taken at which sparks and general noise seen on the pixels becomes noticeable.

The interlock currents at which the HV trips are already shown in tab. D.1. During ramp up of the HV, these trip currents are set higher to avoid trips, while capacitors are being charged.

The gas pressure within the detector was intended to be included into the interlock system to shut down the HV if the pressure inside the detector leaves a certain safe boundary, since this could indicate a leak in the detector or an empty gas bottle.

D.5.2. Gas supply interlock

The gas supply is also part of an interlock system. In case of a window rupture the potential amount of gas that might enter the vacuum system should be limited by electrovalves. Ideally the pressure inside the detector had been included into the gas supply interlock. This could have made sure the gas inlet and outlet are closed in case the pressure inside the detector drops (which might indicate a leak somewhere or an empty gas bottle) or in case of rising pressure.

The latter is not as important though, because a pressure controller is already installed behind the detector, which controls the flow such that the pressure stays at 1050 mbar. While a failure of the controller is thinkable, potentially leading to a pressure increase inside the detector, it is questionable whether this could be dealt with using this interlock system. That is because the pressure sensor used is part of the pressure controller.

Three electrovalves are placed on the gas line of the detector. One, V_{in}, is outside of the building next to the gas bottles (see fig. D.9(a)). The second valve $V_{\text{interlock}}$ is located right before the buffer volume, next to the watercooling system, below the shielding platform, fig. D.9(b). The final electrovalve V_{out} is located after the pressure controller on a blue beam, which supports the optical table below the telescope (see fig. D.9(c)). These valves are normally closed, i.e. in case of power loss they automatically close. They are open if a voltage is applied to them.

The valves are connected to the pressure sensor P_{MM} (see fig. D.2). The pressures to activate the interlock system is defined by upper and lower thresholds asymmetrically. They are as follows:

$$P_{\text{MM, Gas enable}} \leq 9.9 \times 10^{-3}\,\text{mbar}$$

and

$$P_{\text{MM, Gas disable}} \leq 2 \times 10^{-2}\,\text{mbar}$$

(a) Outside V_{in} (b) Below $V_{\text{interlock}}$ (c) Output V_{out}

Figure D.9.: (a): Location of the electrovalve V_{in} located outside the building.(a): Location of the electrovalve $V_{\text{interlock}}$ next to the watercooling system below the shielding table.(a): Location of the electrovalve V_{out} connected to the beam supporting the optical table, on which the telescope is mounted.

D.5.3. CAST magnet interlock

The main CAST magnet interlock, as it is relevant to our detector, is as follows. The gate valve VT3 separating the magnet volume from the telescope volume is interlocked. Only if the vacuum in the telescope volume is good enough, VT3 can be opened while the interlock is activated.

For this the pressure of P_3 is considered relevant (cmp. fig. D.2). The upper and lower thresholds which activate and deactivate the interlock are asymmetric and as follows:

$$P_{3,\,\text{VT3 enable}} \leq 10^{-5}\,\text{mbar}$$

while

$$P_{3,\,\text{VT3 disable}} \leq 8 \times 10^{-5}\,\text{mbar}.$$

This is to make sure there can be no rapid toggling between the two states during pumping or flushing the system.

D.6. CAST log files

The CAST experiment is controlled by a central computer running a slow control system written in LabVIEW [4]. It receives data from all sensors installed at CAST (vacuum pressures, temperatures,

[4] https://www.ni.com/labview

magnet current, magnet position and so on). A separate LabVIEW program is responsible for controlling the magnet position, either by moving the magnet to a specific coordinate or by following the Sun, if it is in reach of the magnet.

The slow control software records all sensor data and presents it in a multitude of graphs. All data is also written to log files. There are two different kinds of log files of interest in the context of the septemboard detector at CAST.

1. The general slow control log file is a versioned space separated values file (similar to a comma separated value (CSV) or tab separated value (TSV) file, except using spaces as delimiters). The filename contains the version number, which decides the columns contained in the file and their order. A `Version.idx` file maps version numbers to file columns for easy access. These slow control log files normally contain one entry every minute. For the Septemboard detector of interest in this file are mainly the magnet current, relevant vacuum pressure sensors and the state of the gate valve `VT3`.

2. The second kind of log files are the tracking log files. These are also space separated value files with fields describing the pointing location of the magnet as well as whether the magnet is currently tracking the Sun.

Both of these log files need to be read by the analysis software to decide whether one or multiple trackings took place in a given background run and if so extract the exact start and end time for a precise data splitting and calculation of background / solar tracking time.

Generally the latter log files are all that is needed to determine the existence of a solar tracking. However, the slow control log file can be used as a further sanity check to make sure the gate valve was actually open and the magnet under current during the solar tracking.

This is implemented in the `cast_log_reader` program part of `TimepixAanlysis`. See sec. O.1.11 for more details on the program.

E. CAST data taking run list — Appendix

The following, table E.1, is a simplified table of the entire run list of data taken at CAST. It includes the run numbers, the type of data (background b and ^{55}Fe calibration c), as they split into each run period. The starting and end times are given for each run. Additionally, the number of trackings in each run as well as the total number of recorded events on the Septemboard and FADC is shown.

Run #	Type	Start	End	Length	# trackings	# frames	# FADC
Run-2							
76	b	*<2017-10-30 Mon 18:39>*	*<2017-11-02 Thu 05:24>*	2 days 10:44	1	88249	19856
77	b	*<2017-11-02 Thu 05:24>*	*<2017-11-03 Fri 05:28>*	1 days 00:03	1	36074	8016
78	b	*<2017-11-03 Fri 05:28>*	*<2017-11-03 Fri 20:45>*	0 days 15:17	1	23506	5988
79	b	*<2017-11-03 Fri 20:46>*	*<2017-11-05 Sun 00:09>*	1 days 03:22	1	40634	8102
80	b	*<2017-11-05 Sun 00:09>*	*<2017-11-05 Sun 23:50>*	0 days 23:40	1	35147	6880
81	b	*<2017-11-05 Sun 23:54>*	*<2017-11-07 Tue 00:00>*	1 days 00:06	1	35856	7283
82	b	*<2017-11-07 Tue 00:01>*	*<2017-11-08 Wed 15:58>*	1 days 15:56	2	59502	12272
83	c	*<2017-11-08 Wed 16:27>*	*<2017-11-08 Wed 17:27>*	0 days 00:59	0	4915	4897
84	b	*<2017-11-08 Wed 17:49>*	*<2017-11-09 Thu 19:01>*	1 days 01:11	1	37391	7551
85	b	*<2017-11-09 Thu 19:01>*	*<2017-11-09 Thu 21:46>*	0 days 02:45	0	4104	899
86	b	*<2017-11-09 Thu 21:47>*	*<2017-11-11 Sat 02:17>*	1 days 04:29	1	42396	9656
87	b	*<2017-11-11 Sat 02:17>*	*<2017-11-12 Sun 14:29>*	1 days 12:11	2	54786	15123
88	c	*<2017-11-12 Sun 14:30>*	*<2017-11-12 Sun 15:30>*	0 days 00:59	0	4943	4934
89	b	*<2017-11-12 Sun 15:30>*	*<2017-11-13 Mon 18:27>*	1 days 02:57	1	25209	6210
90	b	*<2017-11-13 Mon 19:14>*	*<2017-11-14 Tue 20:24>*	1 days 01:09	1	37497	8122
91	b	*<2017-11-14 Tue 20:24>*	*<2017-11-15 Wed 21:44>*	1 days 01:20	1	37732	8108
92	b	*<2017-11-15 Wed 21:45>*	*<2017-11-17 Fri 19:18>*	1 days 21:32	1	67946	14730
93	c	*<2017-11-17 Fri 19:18>*	*<2017-11-17 Fri 20:18>*	0 days 01:00	0	4977	4968
94	b	*<2017-11-17 Fri 20:48>*	*<2017-11-19 Sun 02:34>*	1 days 05:46	1	44344	9422
95	b	*<2017-11-19 Sun 02:35>*	*<2017-11-23 Thu 10:41>*	4 days 08:06	1	154959	33112
96	c	*<2017-11-23 Thu 10:42>*	*<2017-11-23 Thu 17:43>*	0 days 07:01	0	34586	34496
97	b	*<2017-11-23 Thu 17:43>*	*<2017-11-26 Sun 01:41>*	2 days 07:57	1	83404	18277
98	b	*<2017-11-26 Sun 01:42>*	*<2017-11-26 Sun 21:18>*	0 days 19:36	1	29202	6285
99	b	*<2017-11-26 Sun 21:18>*	*<2017-11-28 Tue 06:46>*	1 days 09:27	1	49921	10895
100	b	*<2017-11-28 Tue 06:46>*	*<2017-11-29 Wed 06:40>*	0 days 23:53	1	35658	7841
101	b	*<2017-11-29 Wed 06:40>*	*<2017-11-29 Wed 20:18>*	0 days 13:37	1	20326	4203
102	c	*<2017-11-29 Wed 20:19>*	*<2017-11-29 Wed 22:19>*	0 days 02:00	0	9919	9898
103	b	*<2017-11-29 Wed 22:26>*	*<2017-12-01 Fri 06:46>*	1 days 08:19	1	47381	7867
104	b	*<2017-12-01 Fri 06:47>*	*<2017-12-02 Sat 06:48>*	1 days 00:00	1	35220	5866
105	b	*<2017-12-02 Sat 06:48>*	*<2017-12-03 Sun 06:39>*	0 days 23:51	1	34918	5794
106	b	*<2017-12-03 Sun 06:40>*	*<2017-12-04 Mon 06:54>*	1 days 00:14	1	35576	6018
107	b	*<2017-12-04 Mon 06:54>*	*<2017-12-04 Mon 13:38>*	0 days 06:44	1	9883	1641

Continued on next page

Continued from previous page

Run #	Type	Start	End	Length	# trackings	# frames	# FADC
108	c	<2017-12-04 Mon 13:39>	<2017-12-04 Mon 17:39>	0 days 04:00	0	19503	19448
109	b	<2017-12-04 Mon 17:47>	<2017-12-05 Tue 11:20>	0 days 17:32	1	28402	8217
110	c	<2017-12-05 Tue 11:20>	<2017-12-05 Tue 13:20>	0 days 01:59	0	9804	9786
111	b	<2017-12-05 Tue 13:23>	<2017-12-05 Tue 16:17>	0 days 02:53	0	4244	644
112	b	<2017-12-06 Wed 14:50>	<2017-12-10 Sun 06:46>	3 days 15:55	2	128931	19607
113	b	<2017-12-10 Sun 06:46>	<2017-12-11 Mon 06:49>	1 days 00:03	1	35100	5174
114	b	<2017-12-11 Mon 06:50>	<2017-12-11 Mon 18:33>	0 days 11:43	1	17111	2542
115	b	<2017-12-11 Mon 18:36>	<2017-12-12 Tue 20:58>	1 days 02:21	1	40574	9409
116	c	<2017-12-12 Tue 20:59>	<2017-12-12 Tue 22:59>	0 days 02:00	0	9741	9724
117	b	<2017-12-12 Tue 23:56>	<2017-12-13 Wed 21:29>	0 days 21:33	1	31885	5599
118	c	<2017-12-13 Wed 21:30>	<2017-12-13 Wed 23:30>	0 days 02:00	0	9771	9748
119	b	<2017-12-14 Thu 00:07>	<2017-12-14 Thu 17:04>	0 days 16:57	1	25434	4903
120	c	<2017-12-14 Thu 17:04>	<2017-12-14 Thu 21:04>	0 days 04:00	0	19308	19261
121	b	<2017-12-14 Thu 21:07>	<2017-12-15 Fri 19:22>	0 days 22:14	1	33901	6947
122	c	<2017-12-15 Fri 19:22>	<2017-12-16 Sat 01:20>	0 days 05:57	0	29279	29208
123	b	<2017-12-16 Sat 01:21>	<2017-12-17 Sun 01:06>	0 days 23:45	1	34107	3380
124	b	<2017-12-17 Sun 01:06>	<2017-12-19 Tue 02:57>	2 days 01:50	2	71703	7504
125	b	<2017-12-19 Tue 02:57>	<2017-12-19 Tue 16:20>	0 days 13:22	1	19262	1991
126	c	<2017-12-19 Tue 16:21>	<2017-12-19 Tue 19:21>	0 days 02:59	0	14729	14689
127	b	<2017-12-19 Tue 19:27>	<2017-12-22 Fri 00:17>	2 days 04:50	1	75907	7663
128	c	<2017-12-22 Fri 00:18>	<2017-12-22 Fri 09:23>	0 days 09:05	0	44806	44709
145	c	<2018-02-17 Sat 17:18>	<2018-02-17 Sat 20:40>	0 days 03:22	0	16797	16796
146	b	<2018-02-17 Sat 20:41>	<2018-02-18 Sun 18:12>	0 days 21:30	1	32705	3054
147	c	<2018-02-18 Sun 18:12>	<2018-02-18 Sun 20:12>	0 days 01:59	0	10102	10102
148	b	<2018-02-18 Sun 20:46>	<2018-02-19 Mon 17:24>	0 days 20:37	1	31433	3120
149	c	<2018-02-19 Mon 17:25>	<2018-02-19 Mon 19:25>	0 days 02:00	0	9975	9975
150	b	<2018-02-19 Mon 19:53>	<2018-02-20 Tue 17:36>	0 days 21:42	1	33192	3546
151	c	<2018-02-20 Tue 17:36>	<2018-02-20 Tue 19:36>	0 days 01:59	0	9907	9907
152	b	<2018-02-20 Tue 21:54>	<2018-02-21 Wed 18:05>	0 days 20:10	1	30809	3319
153	c	<2018-02-21 Wed 18:05>	<2018-02-21 Wed 20:05>	0 days 01:59	0	10103	10102
154	b	<2018-02-21 Wed 21:10>	<2018-02-22 Thu 17:23>	0 days 20:12	1	30891	3426
155	c	<2018-02-22 Thu 17:23>	<2018-02-22 Thu 19:23>	0 days 02:00	0	9861	9861
156	b	<2018-02-23 Fri 06:06>	<2018-02-23 Fri 17:41>	0 days 11:35	1	17686	1866
157	c	<2018-02-23 Fri 17:41>	<2018-02-23 Fri 19:41>	0 days 01:59	0	9962	9962
158	b	<2018-02-23 Fri 19:42>	<2018-02-26 Mon 08:46>	2 days 13:03	1	93205	9893
159	c	<2018-02-26 Mon 08:46>	<2018-02-26 Mon 12:46>	0 days 04:00	0	19879	19878
160	b	<2018-02-26 Mon 14:56>	<2018-03-01 Thu 10:24>	2 days 19:28	1	103145	11415
161	c	<2018-03-01 Thu 10:26>	<2018-03-01 Thu 14:26>	0 days 04:00	0	19944	19943
162	b	<2018-03-01 Thu 17:07>	<2018-03-04 Sun 20:16>	3 days 03:08	3	114590	11897
163	c	<2018-03-04 Sun 20:17>	<2018-03-04 Sun 22:17>	0 days 02:00	0	10093	10093
164	b	<2018-03-04 Sun 22:57>	<2018-03-06 Tue 19:15>	1 days 20:18	2	67456	6488
165	c	<2018-03-06 Tue 19:15>	<2018-03-06 Tue 23:15>	0 days 04:00	0	19882	19879
166	b	<2018-03-07 Wed 00:50>	<2018-03-07 Wed 18:28>	0 days 17:38	1	26859	2565
167	c	<2018-03-07 Wed 18:29>	<2018-03-07 Wed 20:29>	0 days 02:00	0	9938	9938
168	b	<2018-03-07 Wed 20:37>	<2018-03-13 Tue 16:54>	5 days 20:16	0	213545	20669
169	c	<2018-03-13 Tue 16:55>	<2018-03-13 Tue 22:55>	0 days 06:00	0	29874	29874
170	b	<2018-03-13 Tue 23:19>	<2018-03-14 Wed 21:01>	0 days 21:42	1	33098	3269
171	c	<2018-03-14 Wed 21:01>	<2018-03-14 Wed 23:01>	0 days 02:00	0	9999	9999

Continued on next page

Continued from previous page

Run #	Type	Start	End	Length	# trackings	# frames	# FADC
172	b	*<2018-03-14 Wed 23:06>*	*<2018-03-15 Thu 17:57>*	0 days 18:50	1	28649	2773
173	c	*<2018-03-15 Thu 17:59>*	*<2018-03-15 Thu 19:59>*	0 days 01:59	0	9898	9897
174	b	*<2018-03-15 Thu 20:39>*	*<2018-03-16 Fri 16:27>*	0 days 19:48	1	30163	2961
175	c	*<2018-03-16 Fri 16:28>*	*<2018-03-16 Fri 18:28>*	0 days 01:59	0	10075	10075
176	b	*<2018-03-16 Fri 18:35>*	*<2018-03-17 Sat 20:55>*	1 days 02:19	1	40084	3815
177	c	*<2018-03-17 Sat 20:55>*	*<2018-03-17 Sat 22:55>*	0 days 01:59	0	9967	9966
178	b	*<2018-03-17 Sat 23:31>*	*<2018-03-22 Thu 17:40>*	4 days 18:09	5	174074	17949
179	c	*<2018-03-22 Thu 17:41>*	*<2018-03-22 Thu 19:41>*	0 days 01:59	0	9887	9887
180	b	*<2018-03-22 Thu 20:47>*	*<2018-03-24 Sat 18:10>*	1 days 21:22	1	69224	7423
181	c	*<2018-03-24 Sat 18:10>*	*<2018-03-24 Sat 22:10>*	0 days 04:00	0	20037	20036
182	b	*<2018-03-24 Sat 23:32>*	*<2018-03-26 Mon 19:46>*	1 days 19:14	2	65888	6694
183	c	*<2018-03-26 Mon 19:47>*	*<2018-03-26 Mon 23:47>*	0 days 03:59	0	20026	20026
184	b	*<2018-03-27 Tue 00:32>*	*<2018-03-30 Fri 14:18>*	3 days 13:45	0	130576	12883
185	c	*<2018-03-30 Fri 14:18>*	*<2018-03-30 Fri 18:18>*	0 days 03:59	0	19901	19901
186	b	*<2018-03-30 Fri 19:03>*	*<2018-04-11 Wed 16:03>*	11 days 21:00	0	434087	42830
187	c	*<2018-04-11 Wed 16:04>*	*<2018-04-11 Wed 20:04>*	0 days 04:00	0	19667	19665
188	b	*<2018-04-11 Wed 20:53>*	*<2018-04-17 Tue 10:53>*	5 days 14:00	0	204281	20781
Run-3							
239	c	*<2018-10-20 Sat 18:31>*	*<2018-10-20 Sat 20:31>*	0 days 02:00	0	9565	9518
240	b	*<2018-10-21 Sun 14:54>*	*<2018-10-22 Mon 16:15>*	1 days 01:21	1	38753	4203
241	c	*<2018-10-22 Mon 16:16>*	*<2018-10-22 Mon 18:16>*	0 days 02:00	0	9480	9426
242	b	*<2018-10-22 Mon 18:44>*	*<2018-10-23 Tue 22:08>*	1 days 03:24	1	41933	4843
243	c	*<2018-10-23 Tue 22:09>*	*<2018-10-24 Wed 00:09>*	0 days 01:59	0	9488	9429
244	b	*<2018-10-24 Wed 00:32>*	*<2018-10-24 Wed 19:24>*	0 days 18:52	1	28870	3317
245	c	*<2018-10-24 Wed 19:25>*	*<2018-10-24 Wed 21:25>*	0 days 01:59	0	9573	9530
246	b	*<2018-10-24 Wed 21:59>*	*<2018-10-25 Thu 16:18>*	0 days 18:18	1	27970	2987
247	c	*<2018-10-25 Thu 16:19>*	*<2018-10-25 Thu 18:19>*	0 days 01:59	0	9389	9334
248	b	*<2018-10-25 Thu 18:25>*	*<2018-10-26 Fri 22:29>*	1 days 04:04	1	42871	4544
249	c	*<2018-10-26 Fri 22:30>*	*<2018-10-27 Sat 00:30>*	0 days 02:00	0	9473	9431
250	b	*<2018-10-27 Sat 01:31>*	*<2018-10-27 Sat 22:26>*	0 days 20:54	1	31961	3552
251	c	*<2018-10-27 Sat 22:26>*	*<2018-10-28 Sun 00:26>*	0 days 01:59	0	9551	9503
253	c	*<2018-10-28 Sun 19:18>*	*<2018-10-28 Sun 21:39>*	0 days 02:20	0	11095	11028
254	b	*<2018-10-28 Sun 21:40>*	*<2018-10-29 Mon 23:03>*	1 days 01:23	1	38991	4990
255	c	*<2018-10-29 Mon 23:03>*	*<2018-10-30 Tue 01:03>*	0 days 02:00	0	9378	9330
256	b	*<2018-10-30 Tue 01:49>*	*<2018-10-31 Wed 22:18>*	1 days 20:29	1	68315	8769
257	c	*<2018-10-31 Wed 22:19>*	*<2018-11-01 Thu 00:19>*	0 days 01:59	0	9648	9592
258	b	*<2018-11-01 Thu 00:20>*	*<2018-11-01 Thu 16:15>*	0 days 15:55	1	24454	3103
259	c	*<2018-11-01 Thu 16:16>*	*<2018-11-01 Thu 17:31>*	0 days 01:14	0	5900	5864
260	c	*<2018-11-01 Thu 17:39>*	*<2018-11-01 Thu 19:09>*	0 days 01:30	0	7281	7251
261	b	*<2018-11-01 Thu 19:39>*	*<2018-11-04 Sun 15:23>*	2 days 19:43	3	103658	12126
262	c	*<2018-11-04 Sun 15:24>*	*<2018-11-04 Sun 21:24>*	0 days 05:59	0	28810	28681
263	b	*<2018-11-05 Mon 00:35>*	*<2018-11-05 Mon 20:28>*	0 days 19:52	1	30428	3610
264	c	*<2018-11-05 Mon 20:28>*	*<2018-11-05 Mon 22:28>*	0 days 01:59	0	9595	9544
265	b	*<2018-11-05 Mon 22:52>*	*<2018-11-07 Wed 22:14>*	1 days 23:21	1	72514	8429
266	c	*<2018-11-07 Wed 22:14>*	*<2018-11-08 Thu 00:14>*	0 days 01:59	0	9555	9506
267	b	*<2018-11-08 Thu 02:05>*	*<2018-11-08 Thu 06:54>*	0 days 04:48	0	7393	929
268	b	*<2018-11-09 Fri 06:15>*	*<2018-11-09 Fri 17:20>*	0 days 11:04	1	16947	1974
269	c	*<2018-11-09 Fri 17:20>*	*<2018-11-09 Fri 21:20>*	0 days 04:00	0	19382	19302

Continued on next page

Continued from previous page

Run #	Type	Start	End	Length	# trackings	# frames	# FADC
270	b	*<2018-11-09 Fri 21:27>*	*<2018-11-11 Sun 21:02>*	1 days 23:34	2	72756	8078
271	c	*<2018-11-11 Sun 21:03>*	*<2018-11-11 Sun 23:46>*	0 days 02:43	0	13015	12944
272	b	*<2018-11-12 Mon 00:09>*	*<2018-11-14 Wed 19:07>*	2 days 18:58	3	102360	11336
273	c	*<2018-11-14 Wed 19:08>*	*<2018-11-14 Wed 21:08>*	0 days 01:59	0	9535	9471
274	b	*<2018-11-14 Wed 21:28>*	*<2018-11-17 Sat 18:14>*	2 days 20:45	3	105187	12101
275	c	*<2018-11-17 Sat 18:14>*	*<2018-11-17 Sat 20:57>*	0 days 02:43	0	13179	13116
276	b	*<2018-11-17 Sat 22:08>*	*<2018-11-22 Thu 02:26>*	4 days 04:17	2	153954	19640
277	c	*<2018-11-22 Thu 02:26>*	*<2018-11-22 Thu 16:14>*	0 days 13:48	0	66052	65749
278	b	*<2018-11-22 Thu 16:14>*	*<2018-11-23 Fri 10:51>*	0 days 18:36	0	28164	3535
279	b	*<2018-11-24 Sat 10:51>*	*<2018-11-26 Mon 14:58>*	2 days 04:07	2	79848	9677
280	c	*<2018-11-26 Mon 14:59>*	*<2018-11-26 Mon 18:59>*	0 days 04:00	0	19189	19112
281	b	*<2018-11-26 Mon 19:02>*	*<2018-11-28 Wed 18:07>*	1 days 23:04	1	72230	8860
282	c	*<2018-11-28 Wed 18:07>*	*<2018-11-28 Wed 20:51>*	0 days 02:43	0	12924	12860
283	b	*<2018-11-28 Wed 22:31>*	*<2018-12-01 Sat 14:38>*	2 days 16:07	3	98246	11965
284	c	*<2018-12-01 Sat 14:39>*	*<2018-12-01 Sat 18:39>*	0 days 03:59	0	19017	18904
285	b	*<2018-12-01 Sat 19:06>*	*<2018-12-03 Mon 19:39>*	2 days 00:33	2	74405	8887
286	c	*<2018-12-04 Tue 15:57>*	*<2018-12-04 Tue 17:57>*	0 days 02:00	0	9766	9715
287	b	*<2018-12-04 Tue 19:07>*	*<2018-12-05 Wed 15:08>*	0 days 20:01	1	30598	3393
288	c	*<2018-12-05 Wed 17:28>*	*<2018-12-05 Wed 19:28>*	0 days 02:00	0	9495	9443
289	b	*<2018-12-05 Wed 23:07>*	*<2018-12-06 Thu 19:11>*	0 days 20:03	1	30629	3269
290	c	*<2018-12-06 Thu 19:11>*	*<2018-12-06 Thu 21:11>*	0 days 02:00	0	9457	9394
291	b	*<2018-12-06 Thu 23:14>*	*<2018-12-08 Sat 13:39>*	1 days 14:24	2	58602	6133
292	c	*<2018-12-08 Sat 13:39>*	*<2018-12-08 Sat 15:39>*	0 days 02:00	0	9475	9426
293	b	*<2018-12-08 Sat 17:42>*	*<2018-12-10 Mon 21:50>*	2 days 04:07	1	79677	8850
294	c	*<2018-12-10 Mon 21:50>*	*<2018-12-10 Mon 23:50>*	0 days 02:00	0	9514	9467
295	b	*<2018-12-11 Tue 00:54>*	*<2018-12-11 Tue 20:31>*	0 days 19:37	1	29981	3271
296	c	*<2018-12-11 Tue 20:31>*	*<2018-12-11 Tue 22:31>*	0 days 02:00	0	9565	9517
297	b	*<2018-12-12 Wed 00:14>*	*<2018-12-13 Thu 18:30>*	1 days 18:15	2	68124	12530
298	b	*<2018-12-13 Thu 18:39>*	*<2018-12-15 Sat 06:41>*	1 days 12:01	1	53497	0
299	b	*<2018-12-15 Sat 06:43>*	*<2018-12-15 Sat 18:13>*	0 days 11:29	1	17061	0
300	c	*<2018-12-15 Sat 18:38>*	*<2018-12-15 Sat 20:38>*	0 days 02:00	0	9466	9415
301	b	*<2018-12-15 Sat 21:34>*	*<2018-12-17 Mon 14:17>*	1 days 16:43	2	62454	7751
302	c	*<2018-12-17 Mon 14:18>*	*<2018-12-17 Mon 16:18>*	0 days 01:59	0	9616	9577
303	b	*<2018-12-17 Mon 16:52>*	*<2018-12-18 Tue 16:41>*	0 days 23:48	1	36583	4571
304	c	*<2018-12-19 Wed 09:33>*	*<2018-12-19 Wed 11:33>*	0 days 01:59	0	9531	9465
306	b	*<2018-12-20 Thu 06:55>*	*<2018-12-20 Thu 11:53>*	0 days 04:58	1	7546	495

Table E.1.: List of all runs recorded with the Septemboard detector during Run-2 and Run-3 at CAST. The run type is listed as b: background with possible tracking and c: calibration with the ^{55}Fe source.

F. Detector behavior over time Appendix

In section 10.2 we covered the median charge and energies of clusters in the background and calibration data. Here in the appendix we present a few additional plots. First, in sec. F.1 are the equivalent figures to fig. 10.5 for other time periods. The goodness-of-fit tests mentioned in sec. 10.2 for the selection of the best interval length are shown as well. In sec. F.2 we have the equivalent figures to fig. 10.4 for the other run periods.

F.1. Choice of gas gain binning time interval

The following figures show the behavior of the different time intervals for the choice of 'ideal' gas gain time binning for all run periods (not in the sense of Run-2 and Run-3, but those split by significant off time). In addition fig. F.2 shows the results of applying a range of goodness of fit tests to the cluster data (we use a plot and not a table for easier visual parsing).

Note that the repository of this thesis contains even more figures related to this in the `Figs/behavior_over_time` directory.

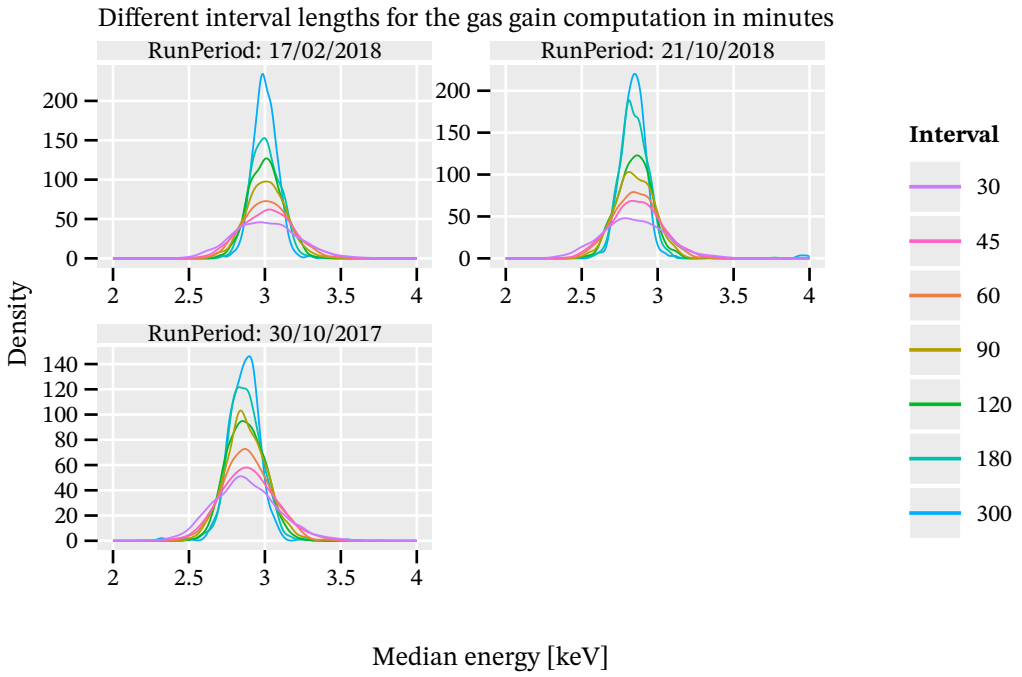


Figure F.1.: Kernel density estimation of the median energies split by the somewhat distinct run periods and the time intervals used. A KDE instead of a histogram is used as the binning has too large of an impact for the dataset.

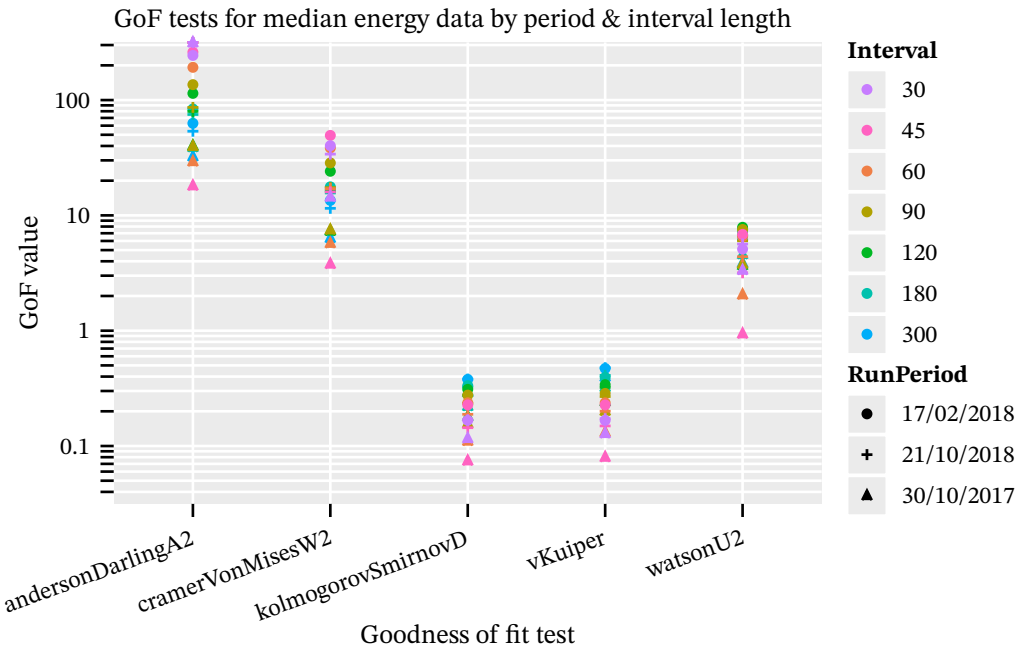


Figure F.2.: Comparison of the different time intervals in each run period using a set of different goodness of fit tests. The 45 min interval seems optimal in the 30/10/2017 period, but worse in others. The 90 min interval is average in most cases.

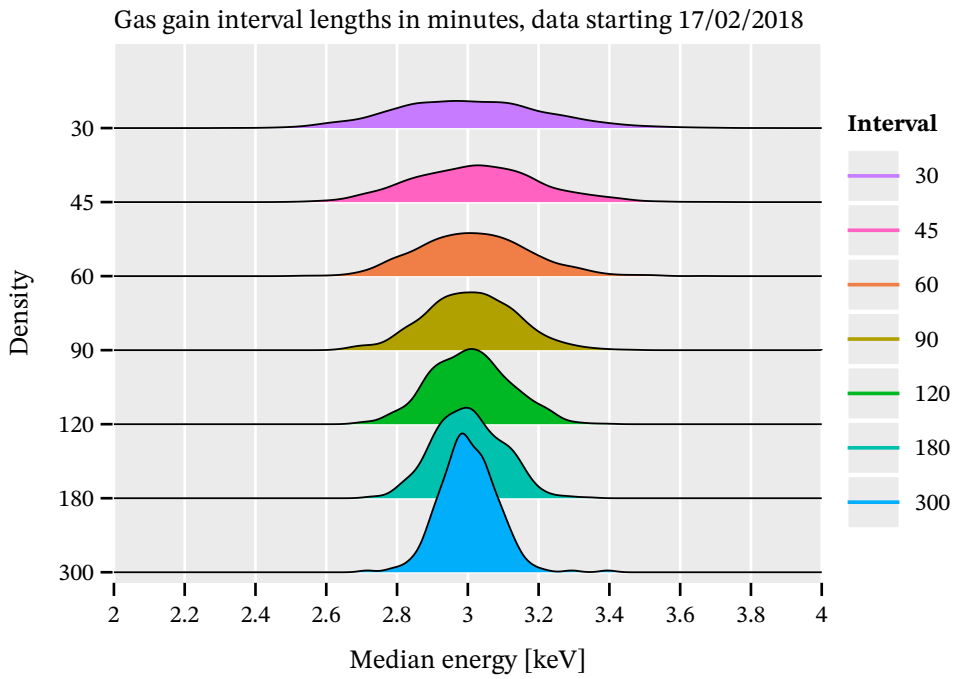


Figure F.3.: Equivalent plot to fig. 10.5 for data from Feb 2018 to Apr 2018. Ridgeline plot of a kernel density estimation (bandwidth based on Silverman's rule of thumb) of the median cluster energies split by the used time intervals. The overlap of the individual ridges is for easier visual comparison and a KDE was selected over a histogram due to strong binning dependence of the resulting histograms.

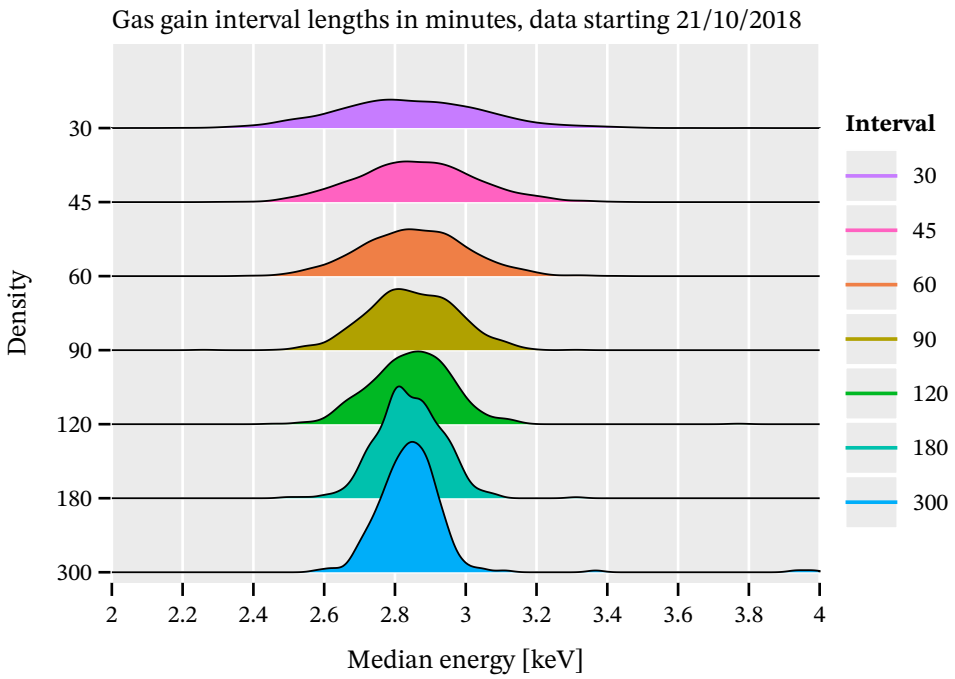


Figure F.4.: Equivalent plot to fig. 10.5 for data from Oct 2018 to Dec 2018. Ridgeline plot of a kernel density estimation (bandwidth based on Silverman's rule of thumb) of the median cluster energies split by the used time intervals. The overlap of the individual ridges is for easier visual comparison and a KDE was selected over a histogram due to strong binning dependence of the resulting histograms.

F.2. Correlation of gas gain and ambient CAST temperature

Let's look at the rest of the data not shown in sec. 10.2.1. First in fig. F.5 we see the same plot as fig. 10.4, but only for Run-2 data from 2017. The anti-correlation is not quite visible here, instead in parts it seems like the expected correlation of temperature and gas gain is visible. Fig. F.6 is the data for Feb 2018 to Apr 2018. Here there seems to be some of the anti correlation, but less than in the Run-3 data presented in the main of the discussion. Finally, fig. F.7 shows the gas gain of the center chip plotted directly against the ambient temperature at CAST as a scatter plot. Here the anti correlation becomes very visible as a global trend.

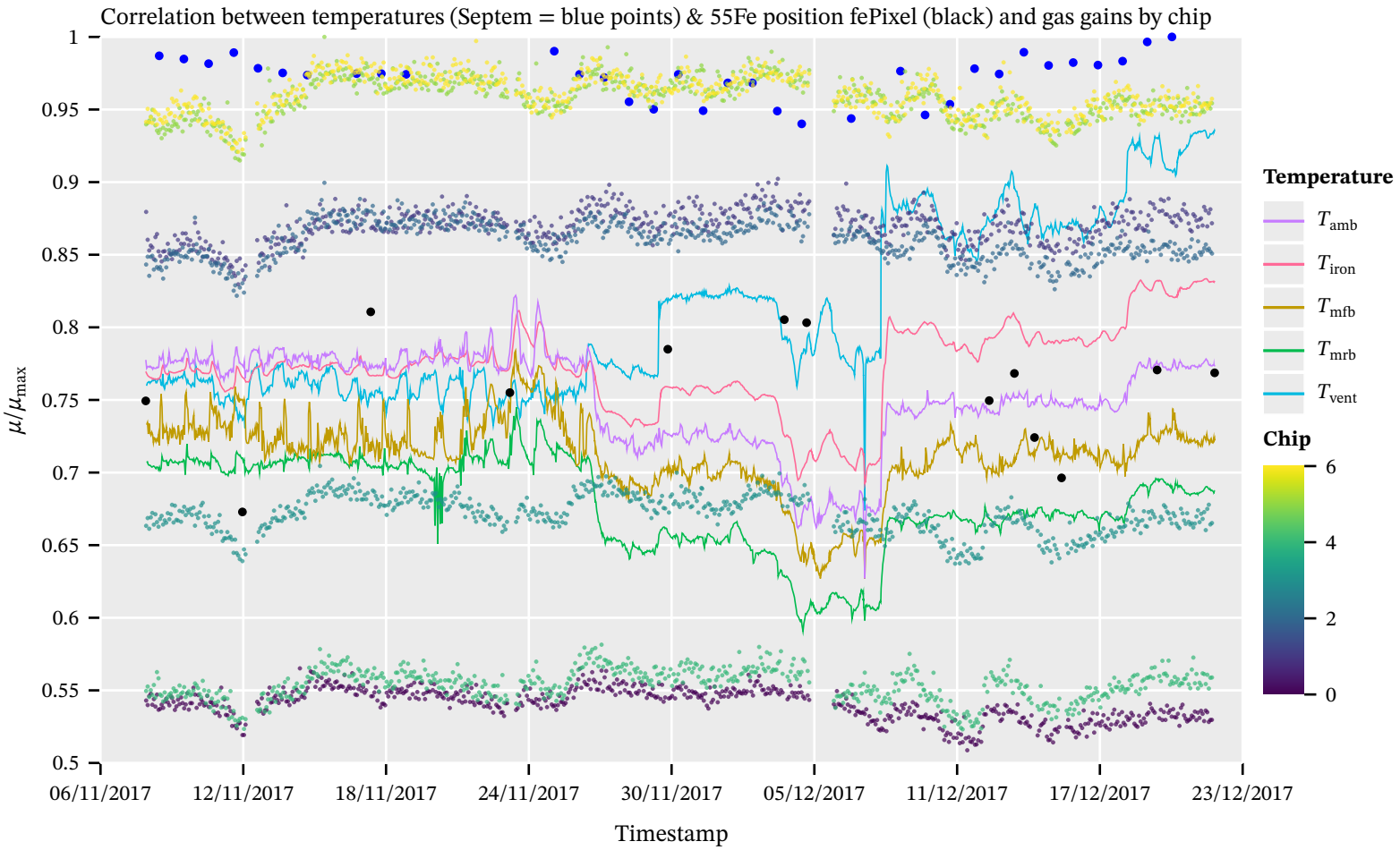


Figure F.5.: Normalized data for Run-2 (only 2017) of the temperature sensors from the CAST slow control log files compared to the behavior of the mean peak position in the ^{55}Fe pixel spectra (black points), the recovered temperature values recorded during each solar tracking (blue points) and the gas gain values computed based on 90 min of data for each chip (smaller points using Viridis color scale). The shift log temperatures nicely follow the trend of the general temperatures. In this period no real anti-correlation is visible. Instead in parts it looks like the expected proportionality between temperature and gas gain appears.

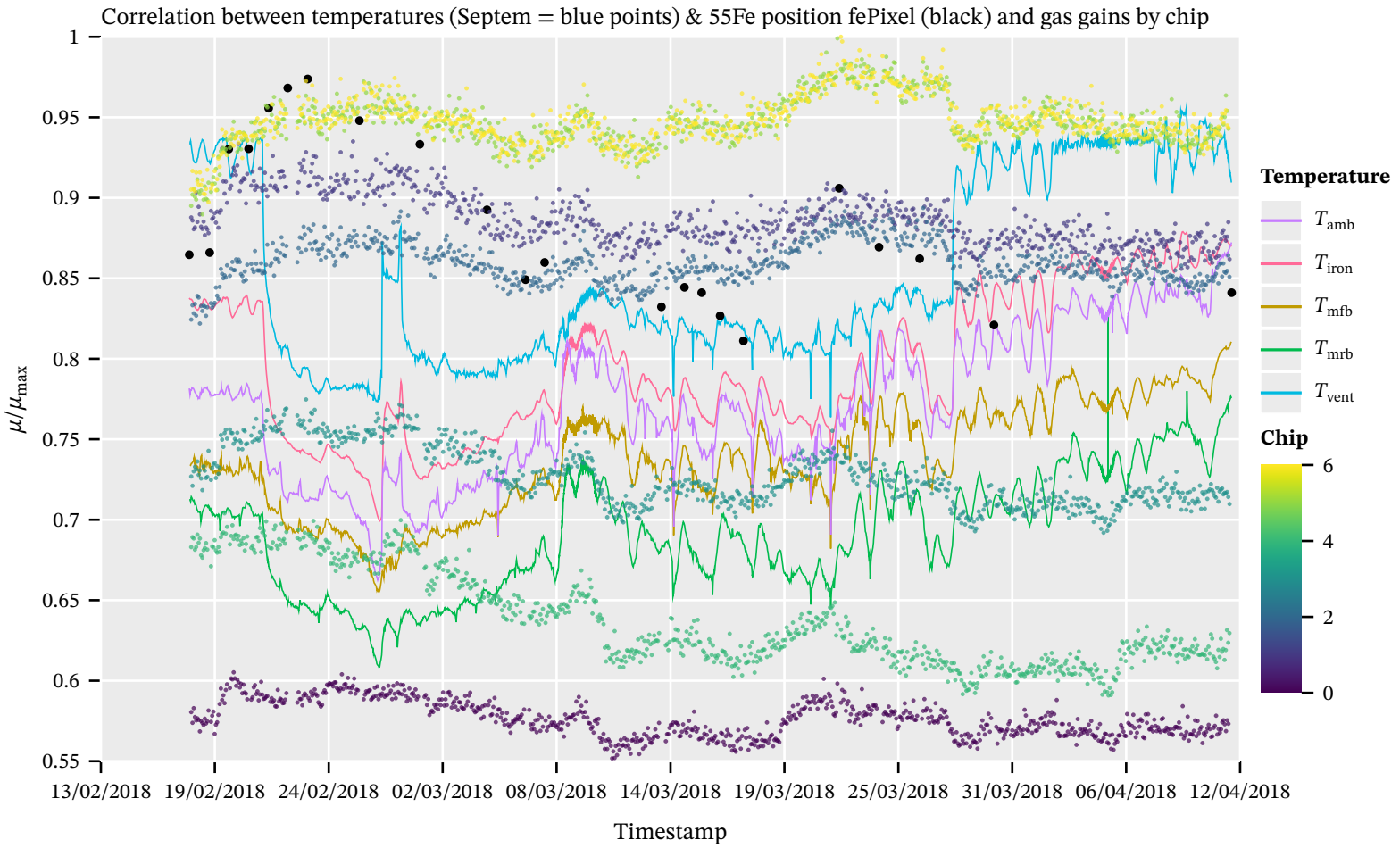


Figure F.6.: Normalized data for Run-2 (only Feb. to Apr. of 2018) of the temperature sensors from the CAST slow control log files compared to the behavior of the mean peak position in the ^{55}Fe pixel spectra (black points), the recovered temperature values recorded during each solar tracking (blue points) and the gas gain values computed based on 90 min of data for each chip (smaller points using Viridis color scale). The shift log temperatures nicely follow the trend of the general temperatures. Here the anti correlation seems to be visible in some parts, but also less extreme than in the end of 2018 Run-3 data, presented in the main section.

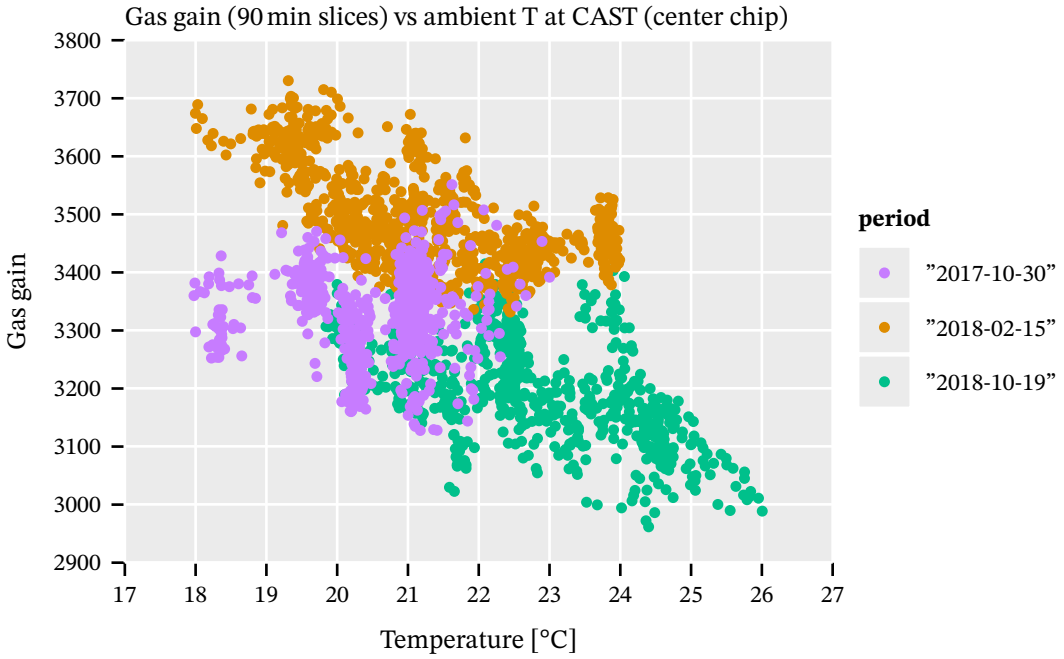


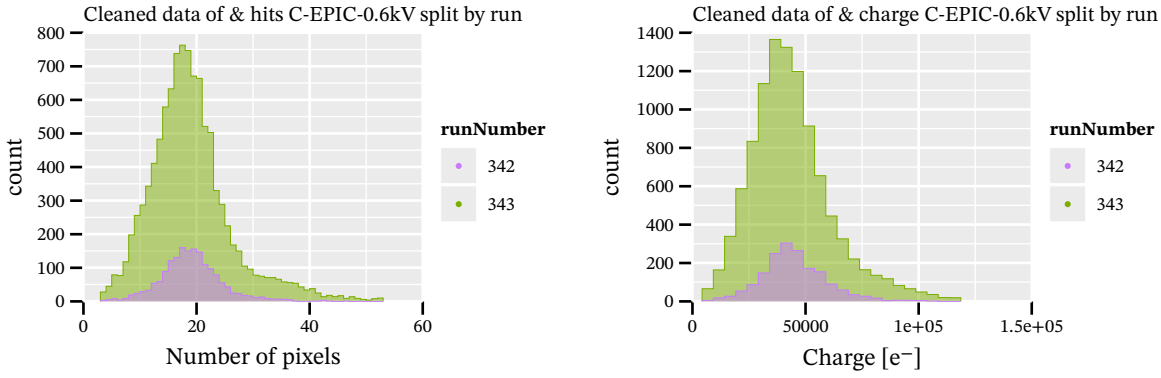
Figure F.7.: Gas gains of the center chip (by 90 min time slices) against the ambient temperature at CAST. As a general trend the anti correlation is very visible. Also visible though is that for the 2017 Run-2 data that effect does not really appear.

G. CAST Detector Lab data — Appendix

In this appendix we present additional plots of the CAST Detector Lab (CDL) data taking campaign, as introduced in sec. 11.2. First in sec. G.1 plots of all CDL runs by their target and filter combination, both for the pixel and charge spectrum, can be found, split by each run. Sec. G.2 then contains all figures equivalent to fig. 11.4 for all the different runs (both as pixel and charge spectrum).

G.1. All spectra split by run

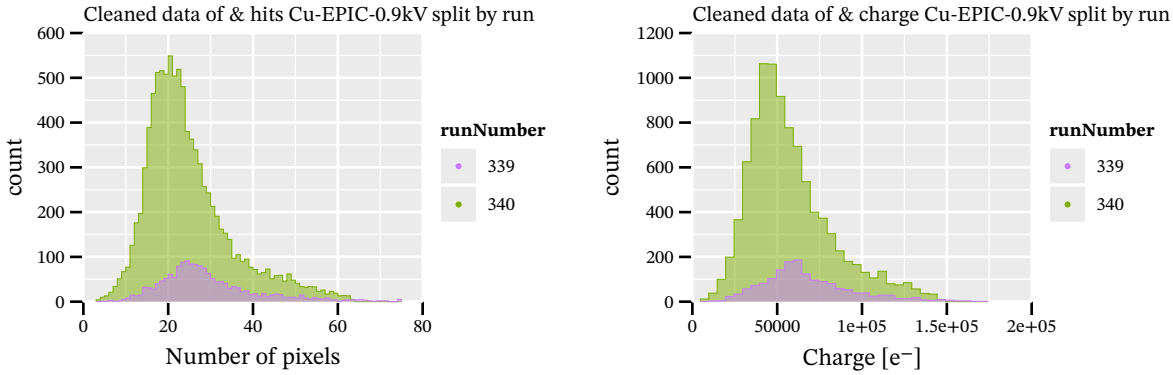
The plots in this section are useful to see the variation visible for one target/filter combination between different runs. In theory the peaks should always be at the same pixel / charge value, but in reality for some targets they vary significantly between runs. This is the main motivator to treat the data on a 'run-by-run' basis.



(a) Pixel spectrum

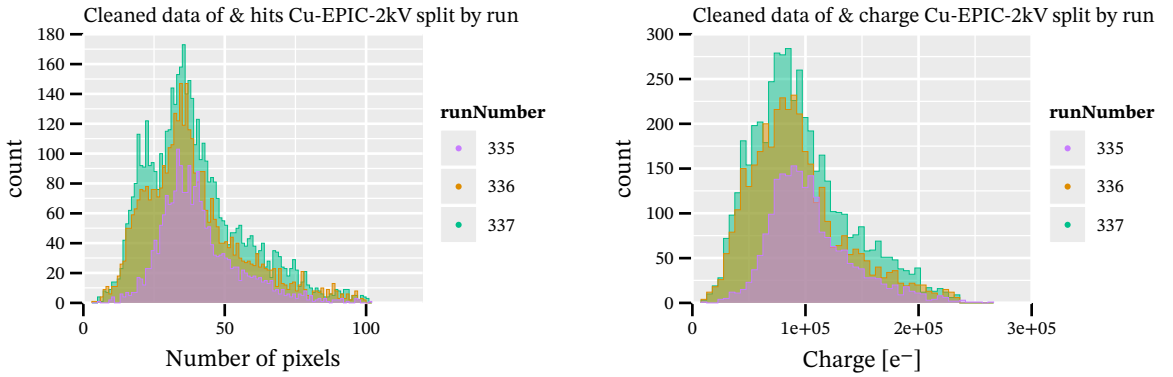
(b) Charge spectrum

Figure G.1.: Pixel spectra, (a), and charge spectra, (b), of the raw data split by the data taking run.



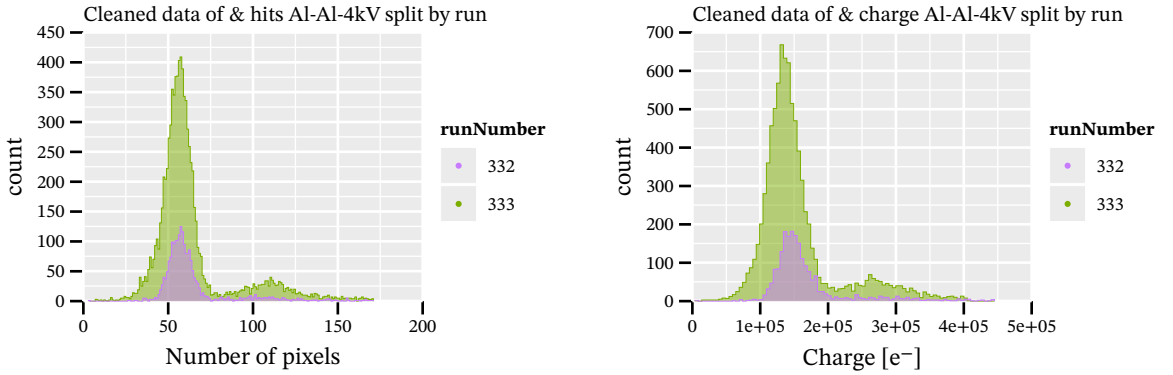
(a) Pixel spectrum (b) Charge spectrum

Figure G.2.: Pixel spectra, (a), and charge spectra, (b), of the raw data split by the data taking run.



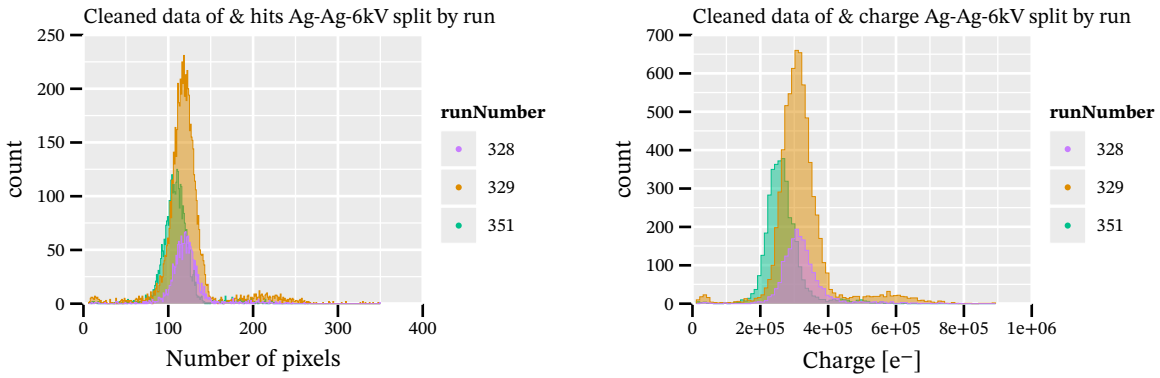
(a) Pixel spectrum (b) Charge spectrum

Figure G.3.: Pixel spectra, (a), and charge spectra, (b), of the raw data split by the data taking run.



(a) Pixel spectrum (b) Charge spectrum

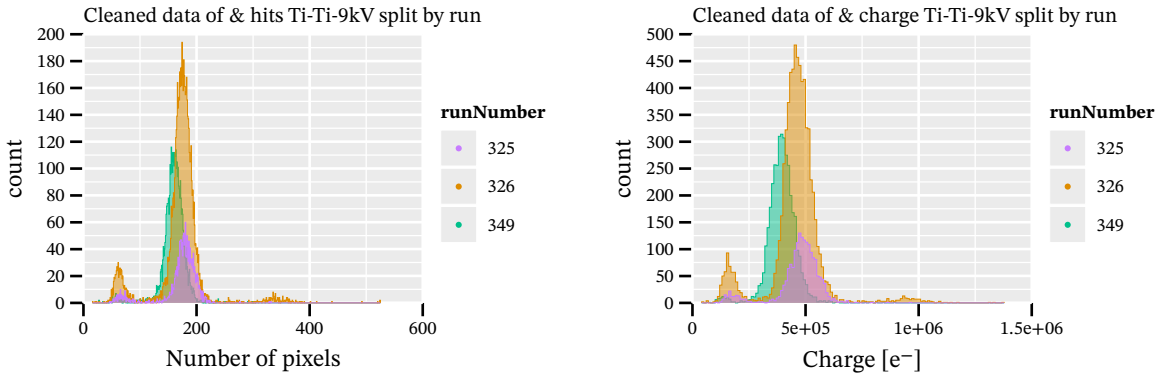
Figure G.4.: Pixel spectra, (a), and charge spectra, (b), of the raw data split by the data taking run.



(a) Pixel spectrum

(b) Charge spectrum

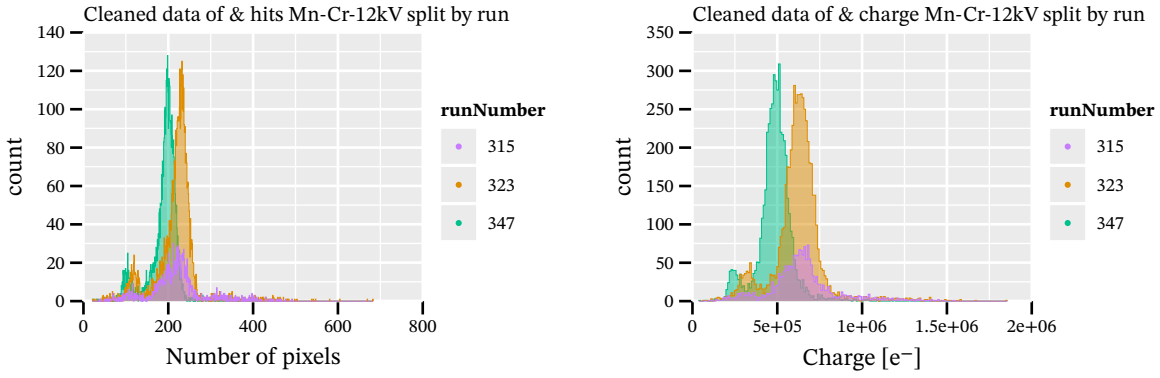
Figure G.5.: Pixel spectra, (a), and charge spectra, (b), of the raw data split by the data taking run.



(a) Pixel spectrum

(b) Charge spectrum

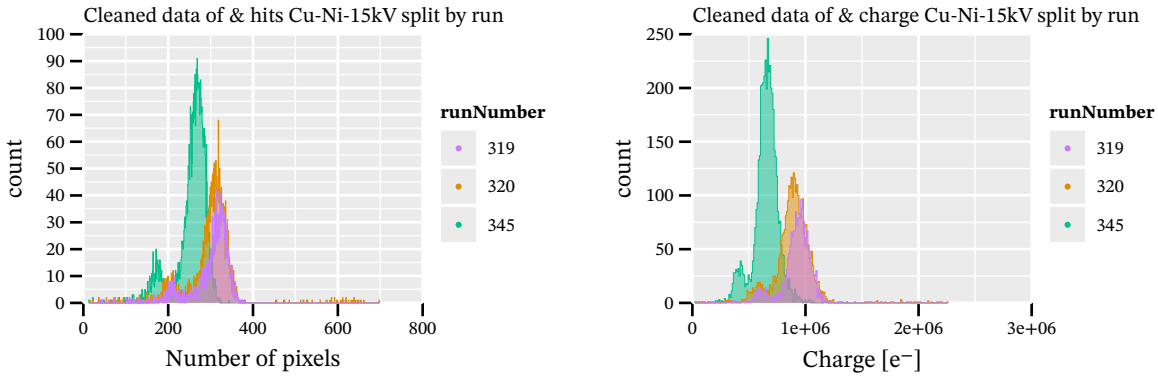
Figure G.6.: Pixel spectra, (a), and charge spectra, (b), of the raw data split by the data taking run.



(a) Pixel spectrum

(b) Charge spectrum

Figure G.7.: Pixel spectra, (a), and charge spectra, (b), of the raw data split by the data taking run.



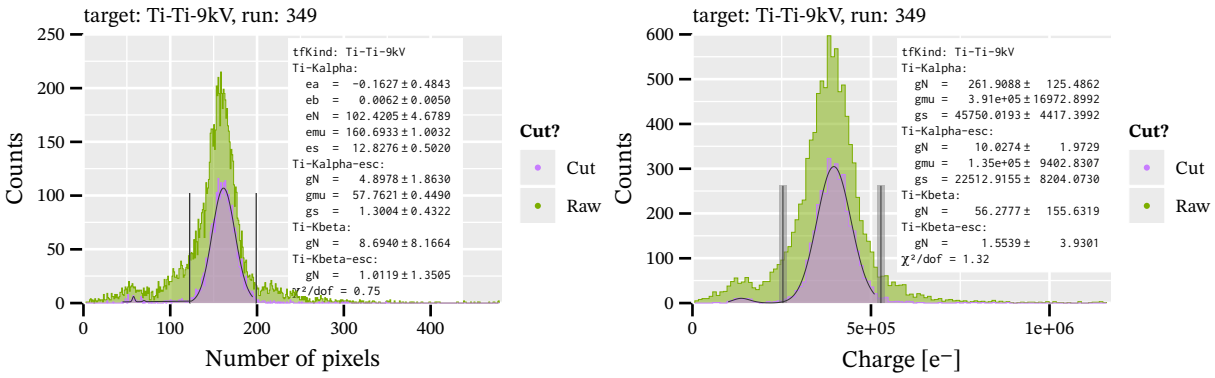
(a) Pixel spectrum

(b) Charge spectrum

Figure G.8.: Pixel spectra, (a), and charge spectra, (b), of the raw data split by the data taking run.

G.2. All CDL spectra with line fits by run

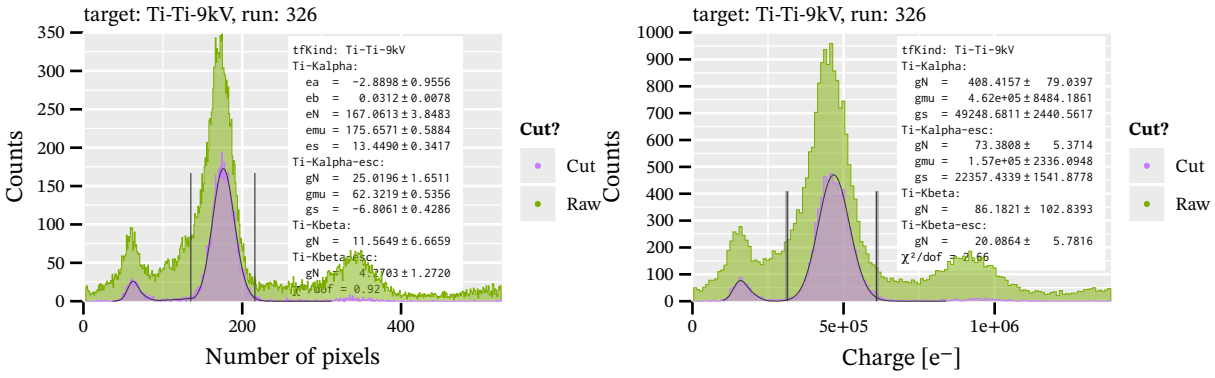
The plots in this section are the raw and cleaned pixel and charge spectra for each target/filter combination, split by run. Any missing parameter indicates it was fixed. For lines with an explicit 'fixed', all parameters were fixed.



(a) Pixel spectrum

(b) Charge spectrum

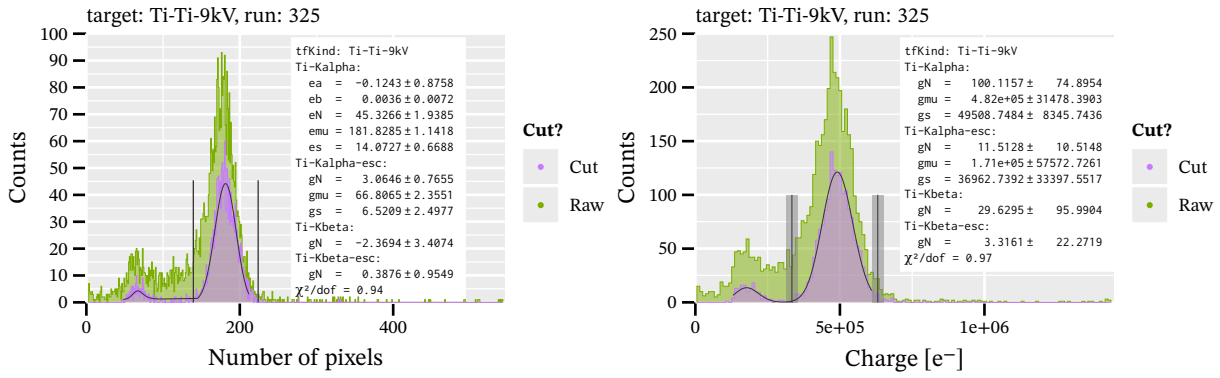
Figure G.9.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Ti-Ti-9kV dataset for run 349.



(a) Pixel spectrum

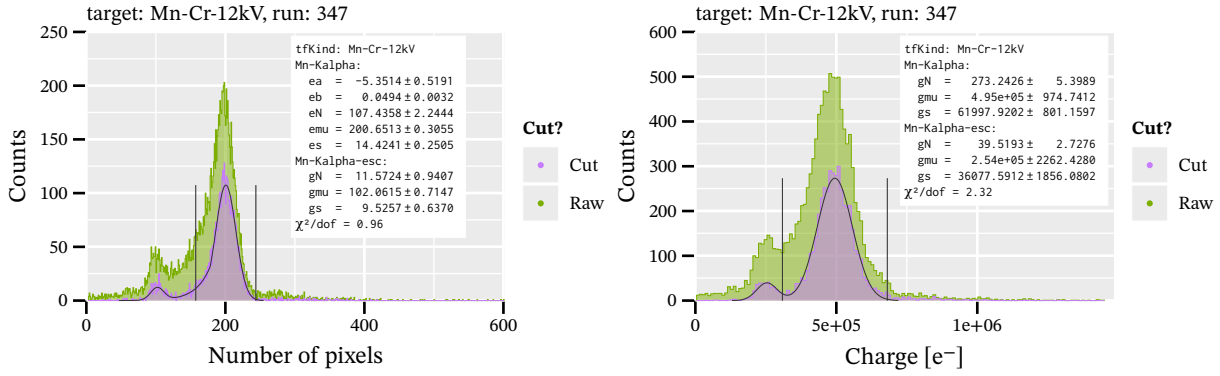
(b) Charge spectrum

Figure G.10.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Ti-Ti-9kV dataset for run 326.



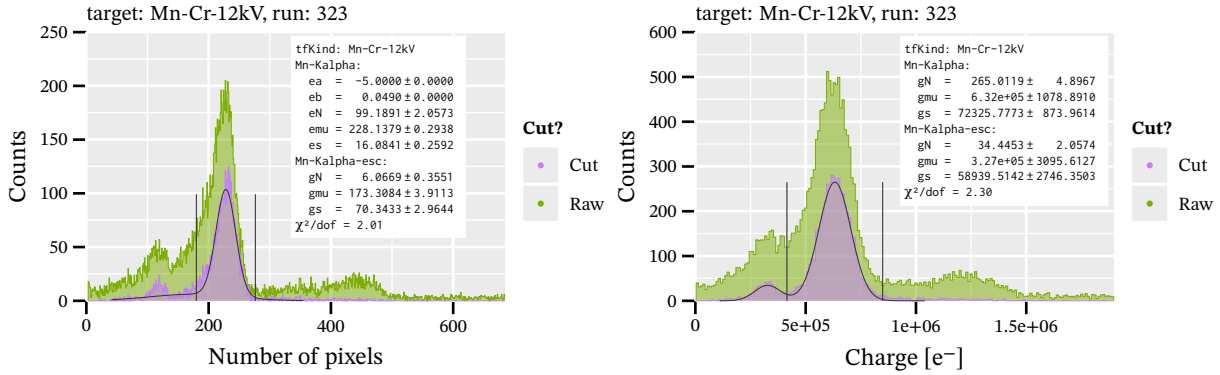
(a) Pixel spectrum (b) Charge spectrum

Figure G.11.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Ti-Ti-9kV dataset for run 325.



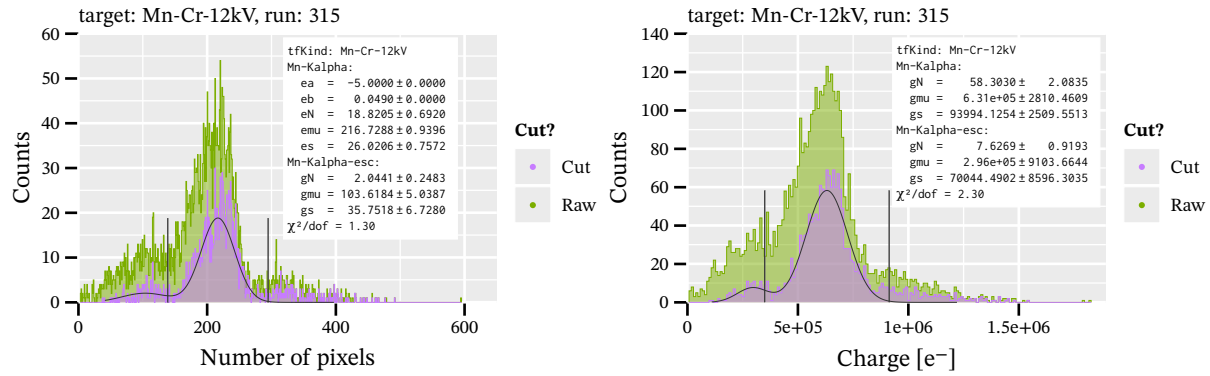
(a) Pixel spectrum (b) Charge spectrum

Figure G.12.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Mn-Cr-12kV dataset for run 347.



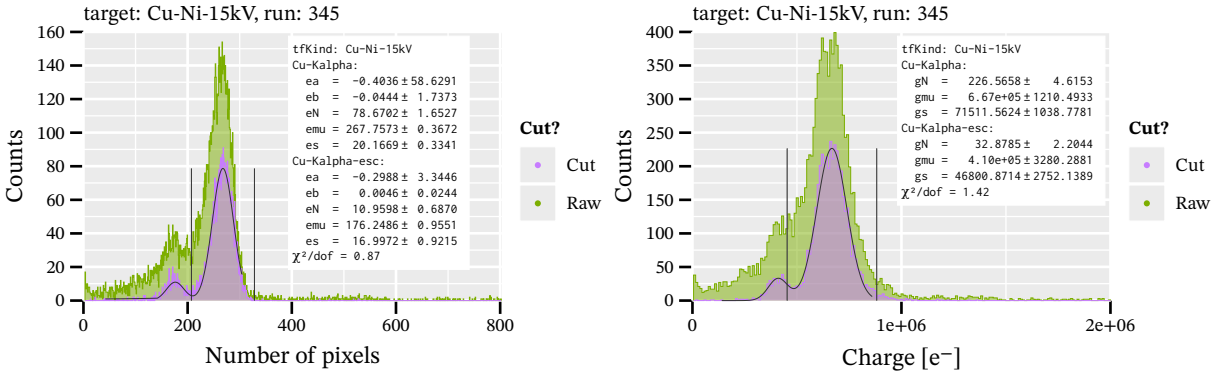
(a) Pixel spectrum (b) Charge spectrum

Figure G.13.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Mn-Cr-12kV dataset for run 323.



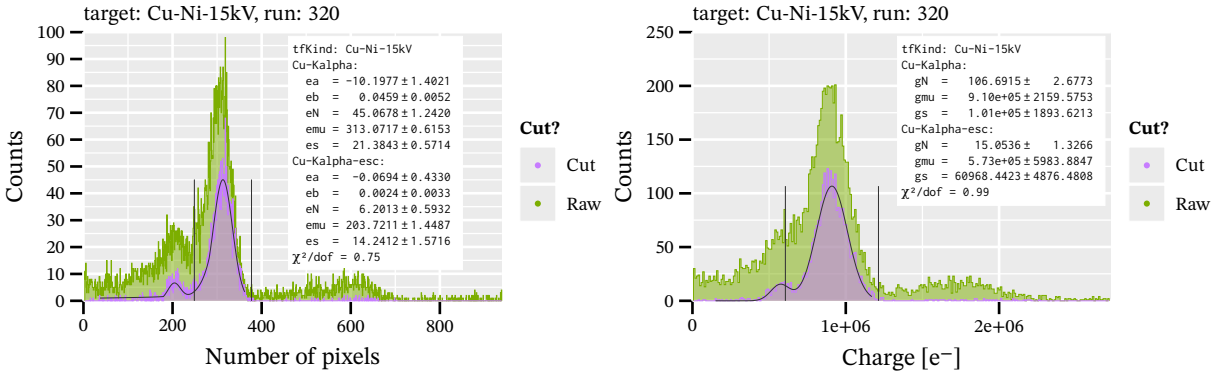
(a) Pixel spectrum (b) Charge spectrum

Figure G.14.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Mn-Cr-12kV dataset for run 315.



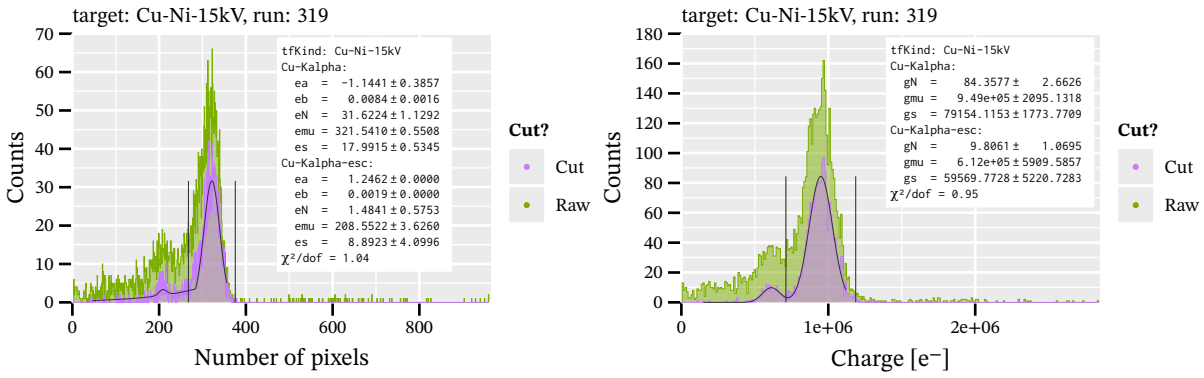
(a) Pixel spectrum (b) Charge spectrum

Figure G.15.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Cu-Ni-15kV dataset for run 345.



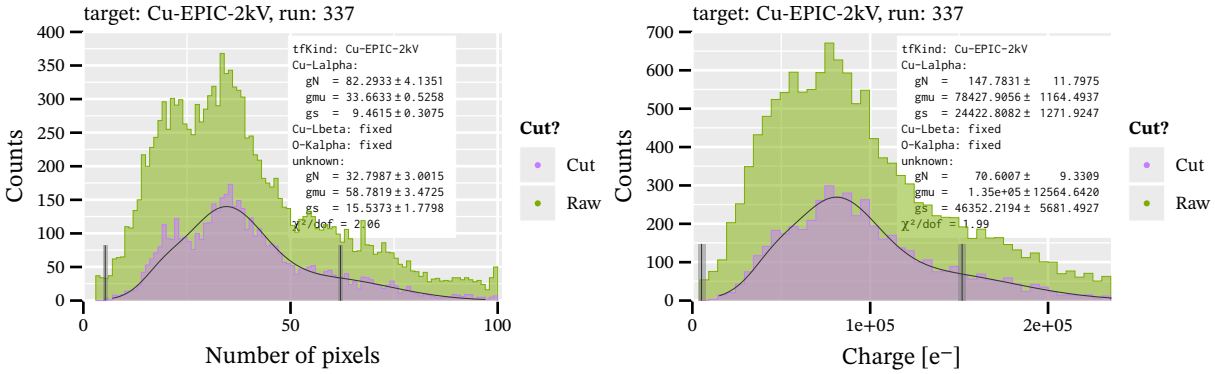
(a) Pixel spectrum (b) Charge spectrum

Figure G.16.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Cu-Ni-15kV dataset for run 320.



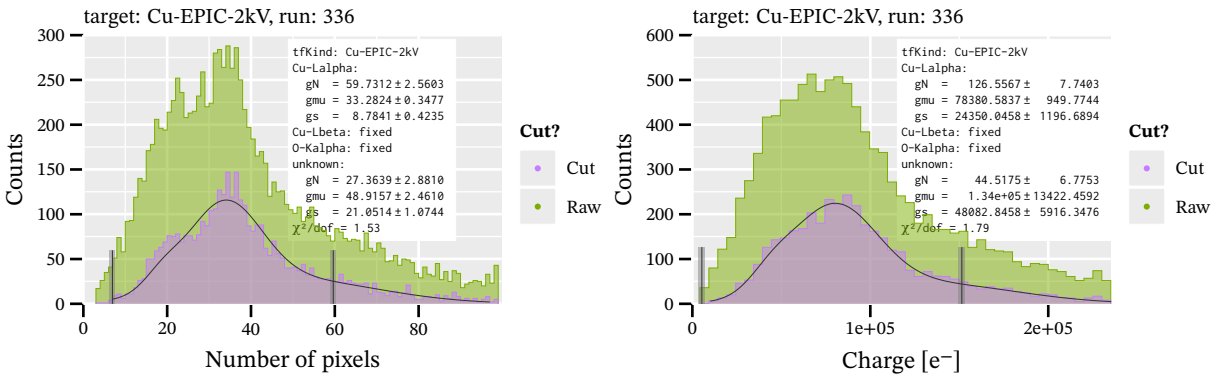
(a) Pixel spectrum (b) Charge spectrum

Figure G.17.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Cu-Ni-15kV dataset for run 319.



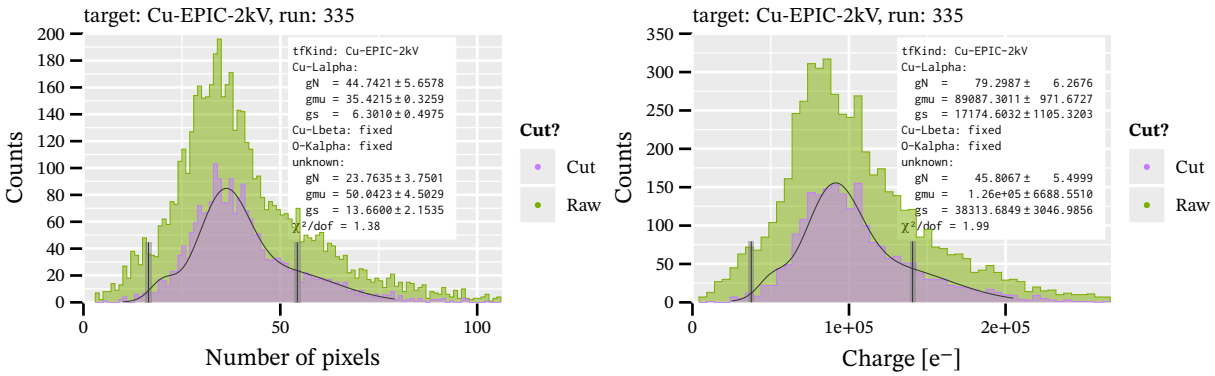
(a) Pixel spectrum (b) Charge spectrum

Figure G.18.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Cu-EPIC-2kV dataset for run 337.



(a) Pixel spectrum (b) Charge spectrum

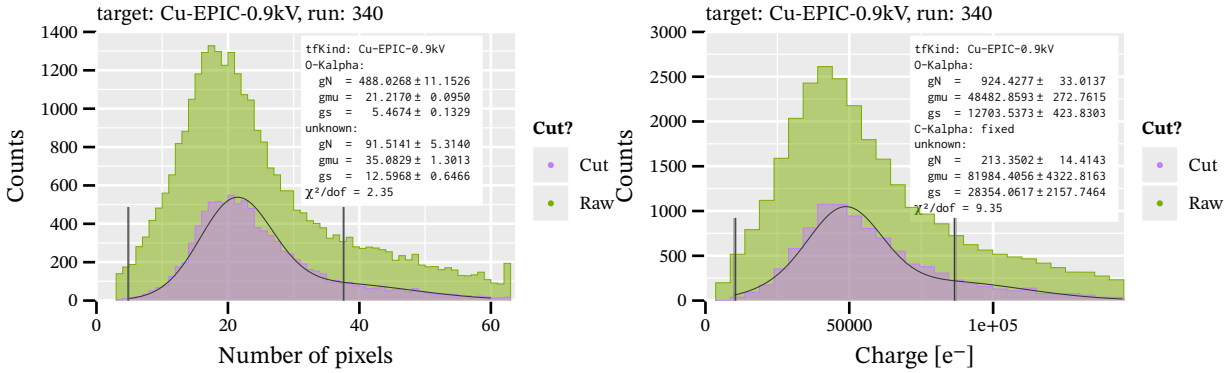
Figure G.19.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Cu-EPIC-2kV dataset for run 336.



(a) Pixel spectrum

(b) Charge spectrum

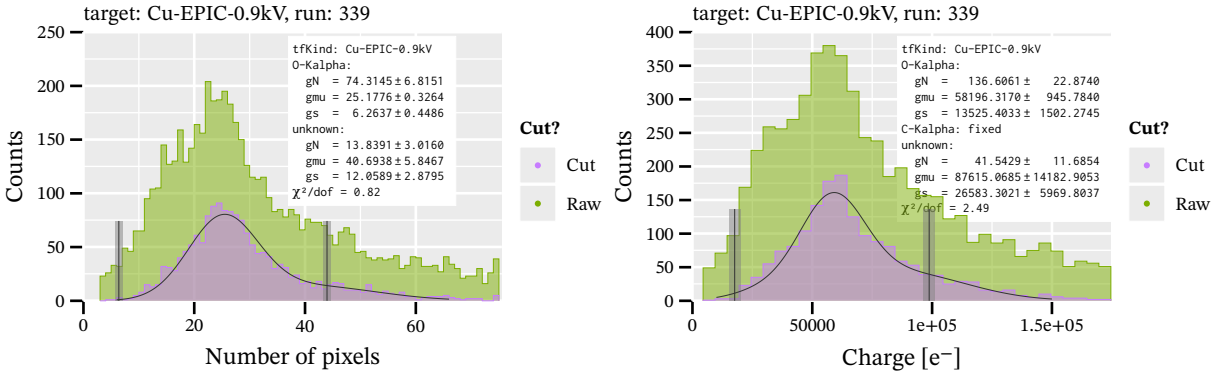
Figure G.20.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Cu-EPIC-2kV dataset for run 335.



(a) Pixel spectrum

(b) Charge spectrum

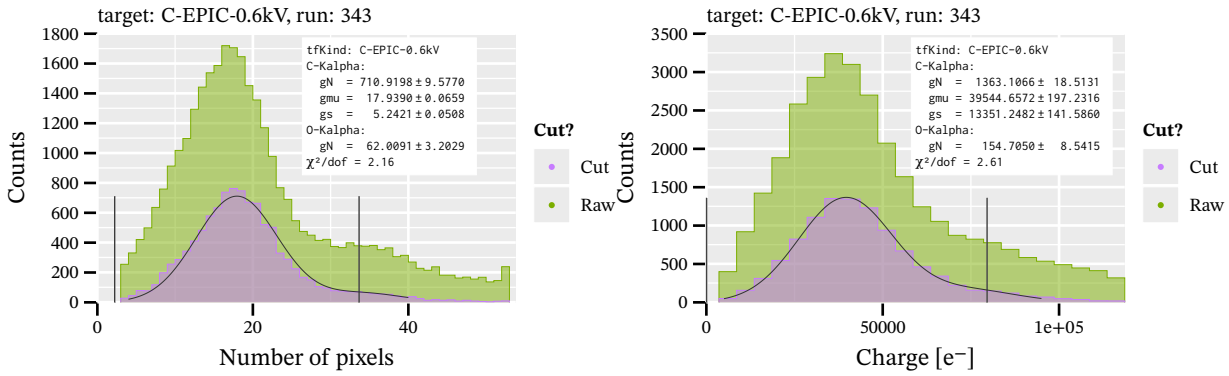
Figure G.21.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Cu-EPIC-0.9kV dataset for run 340.



(a) Pixel spectrum

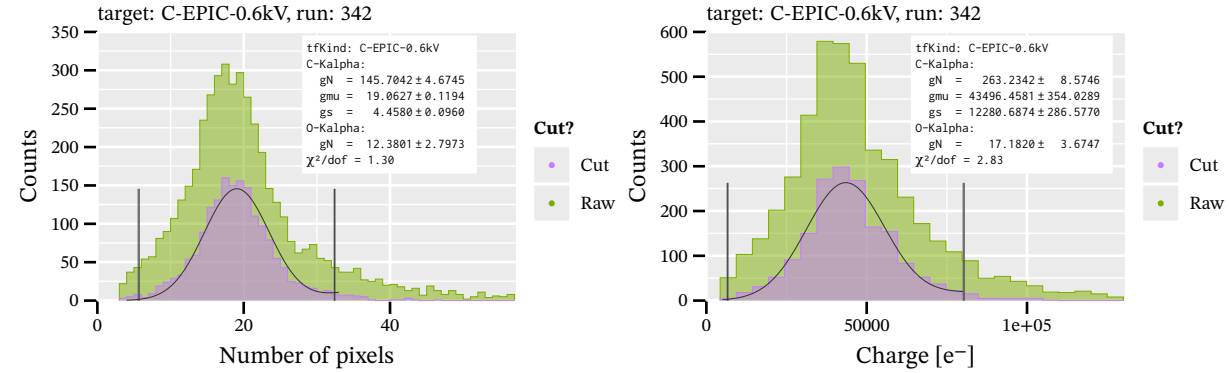
(b) Charge spectrum

Figure G.22.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Cu-EPIC-0.9kV dataset for run 339.



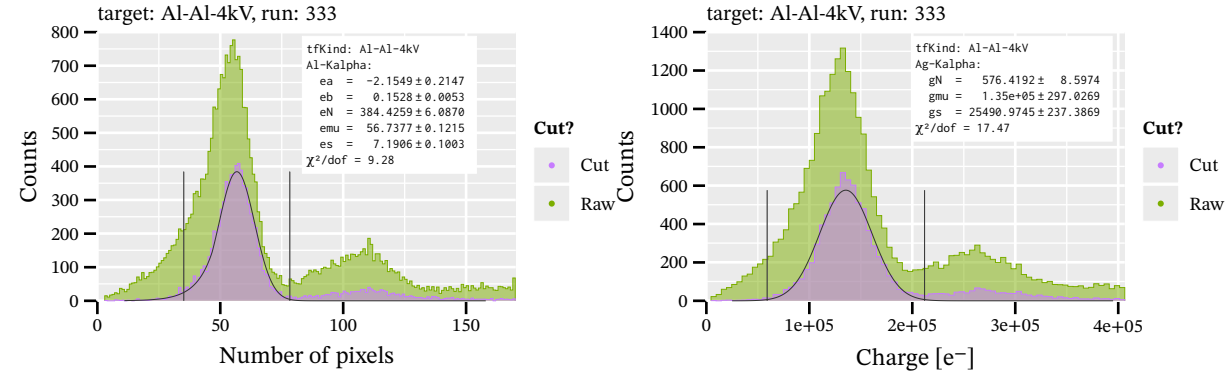
(a) Pixel spectrum (b) Charge spectrum

Figure G.23.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the C-EPIC-0.6kV dataset for run 343.



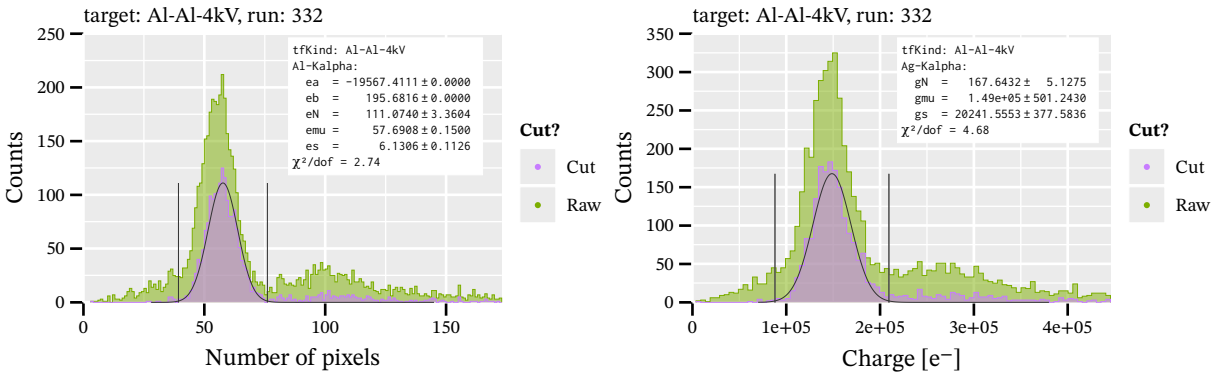
(a) Pixel spectrum (b) Charge spectrum

Figure G.24.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the C-EPIC-0.6kV dataset for run 342.



(a) Pixel spectrum (b) Charge spectrum

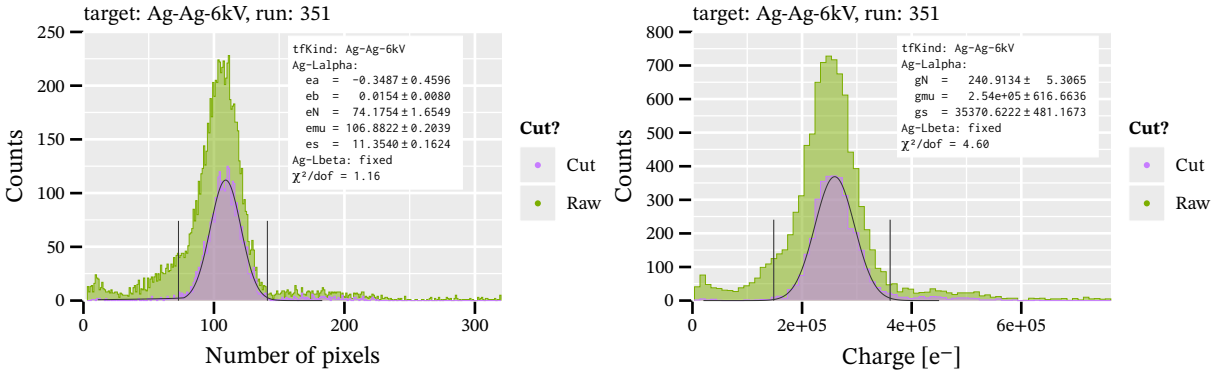
Figure G.25.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Al-Al-4kV dataset for run 333.



(a) Pixel spectrum

(b) Charge spectrum

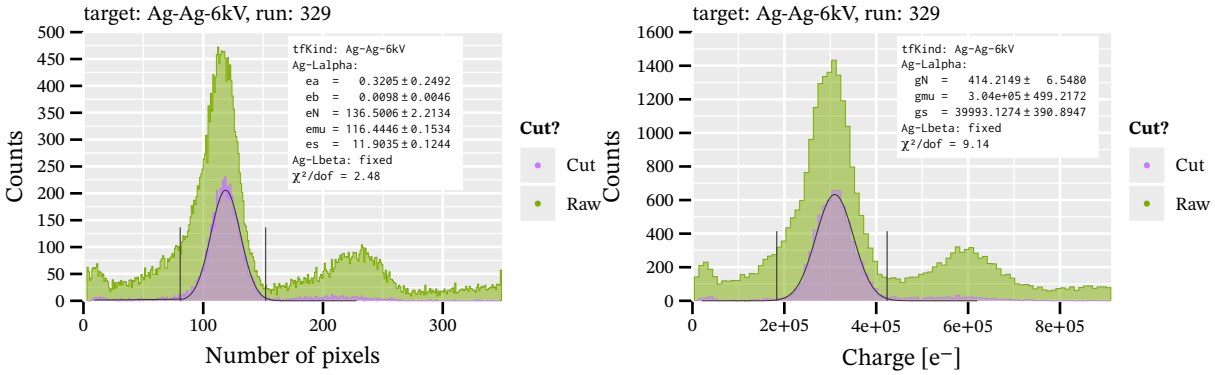
Figure G.26.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Al-Al-4kV dataset for run 332.



(a) Pixel spectrum

(b) Charge spectrum

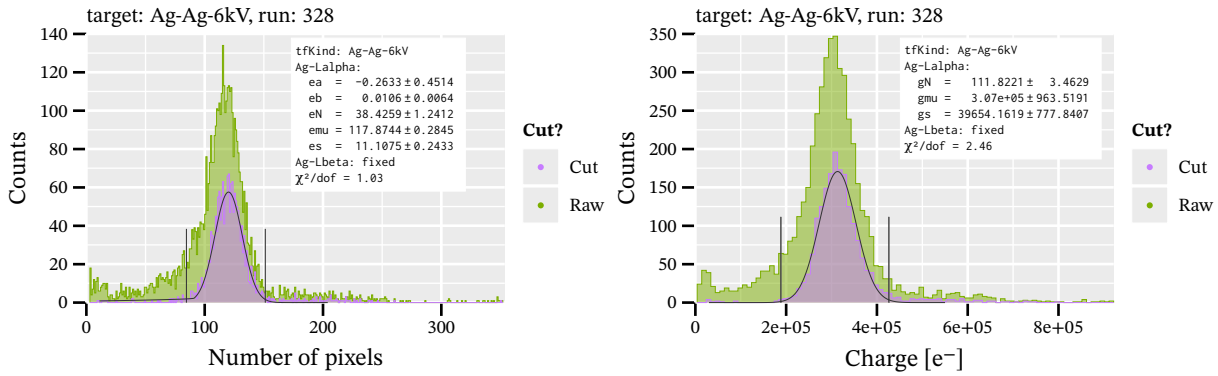
Figure G.27.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Ag-Ag-6kV dataset for run 351.



(a) Pixel spectrum

(b) Charge spectrum

Figure G.28.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Ag-Ag-6kV dataset for run 329.



(a) Pixel spectrum

(b) Charge spectrum

Figure G.29.: Pixel spectrum, (a), and charge spectrum, (b), including fit parameters, raw and cut data of the Ag-Ag-6kV dataset for run 328.

H. CAST Detector Lab variations and fitting by run Appendix

The main difference between how the CAST detector lab data is treated in [134] and in this thesis is that here we treat each data taking run of the CDL data as one unit, instead of all data for one target/filter combination. This is mainly due to the strong detector variation in the gas gain during the CDL data taking. As such it is not possible to fit the spectra when combining all data from all runs for a single target/filter combination. In the extreme cases (Cu − Ni for example) this leads to visibly two main peaks in the charge spectrum. Each run is fit separately. The one question about this approach is whether the cluster properties are stable under gas gain changes. Sec. H.1 shows that this is indeed the case.

H.1. Influence of gas gain variations on cluster properties

The following figures are ridgeline plots of all relevant cluster properties as introduced in sec. 8.4.2. For each plot and each property all CDL runs are shown as kernel density estimations. Outside the number of hits and total charge in a cluster (which are expected to vary with gas gain of course) the properties remain stable even in the cases that vary strongly.

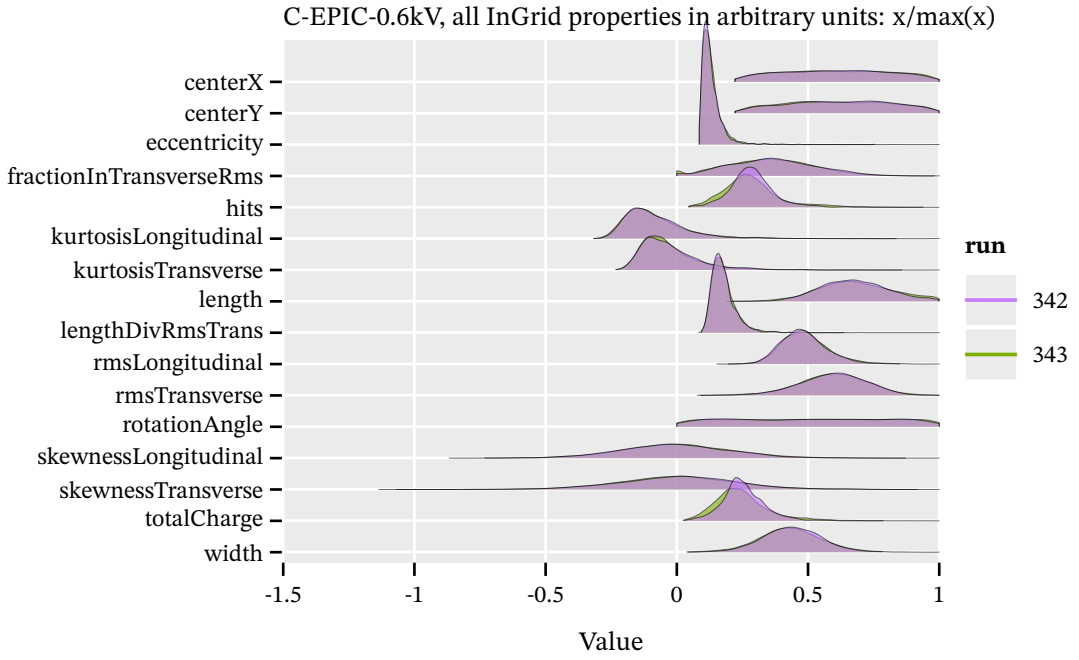


Figure H.1.: Ridgeline plot of kernel density estimations of all cluster properties split by each CDL run. Target/filter: C-EPIC-0.6kV

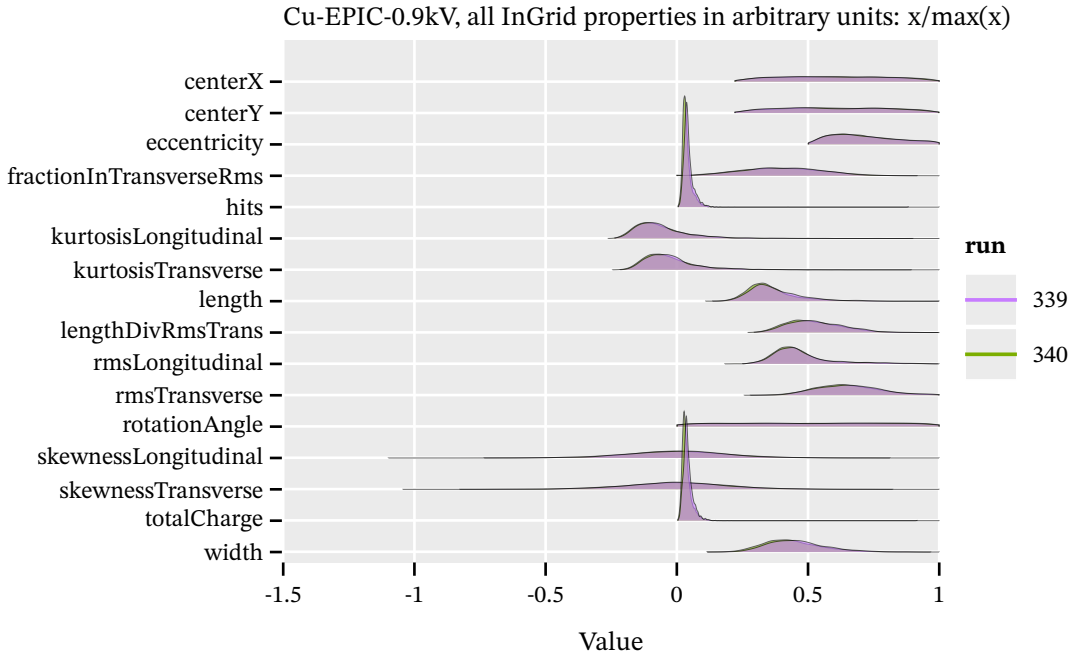


Figure H.2.: Ridgeline plot of kernel density estimations of all cluster properties split by each CDL run. Target/filter: Cu-EPIC-0.9kV

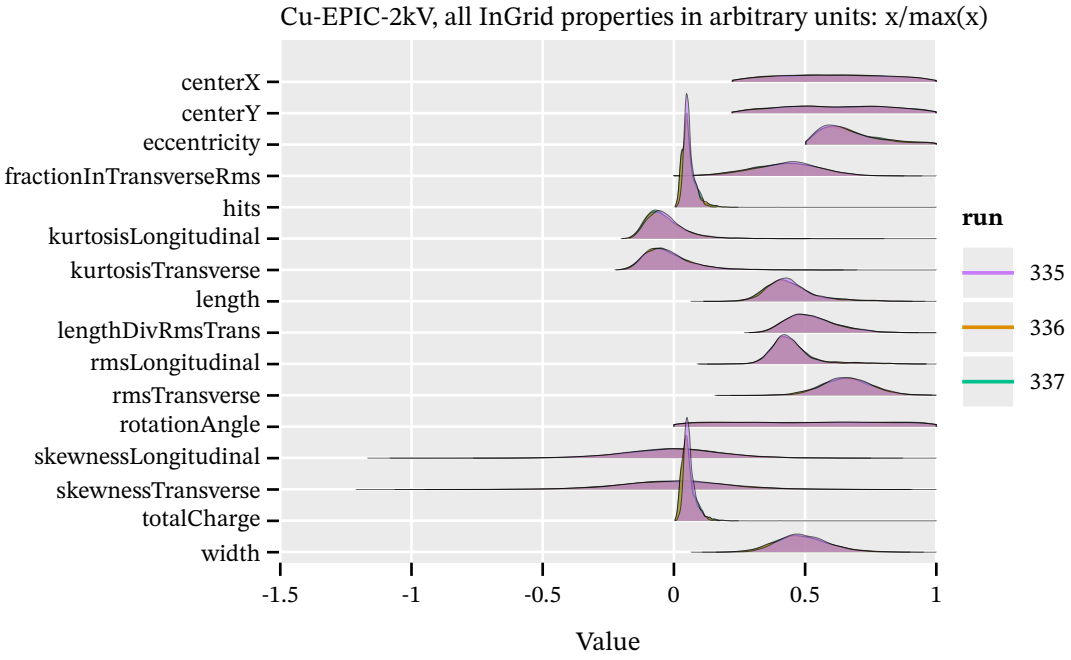


Figure H.3.: Ridgeline plot of kernel density estimations of all cluster properties split by each CDL run. Target/filter: Cu-EPIC-2kV

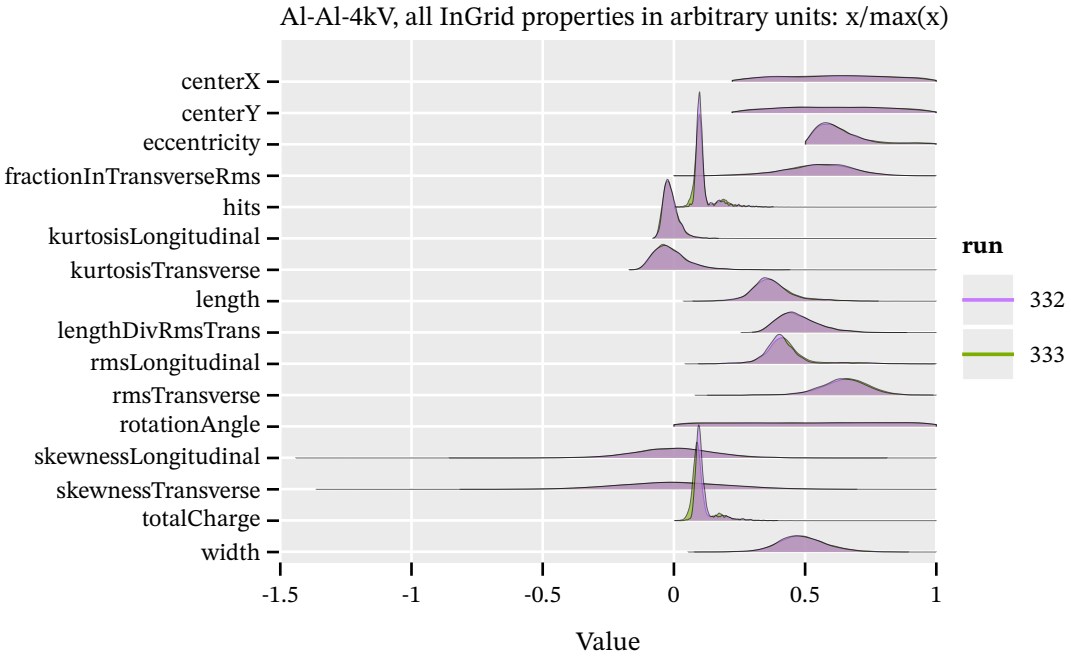


Figure H.4.: Ridgeline plot of kernel density estimations of all cluster properties split by each CDL run. Target/filter: Al-Al-4kV

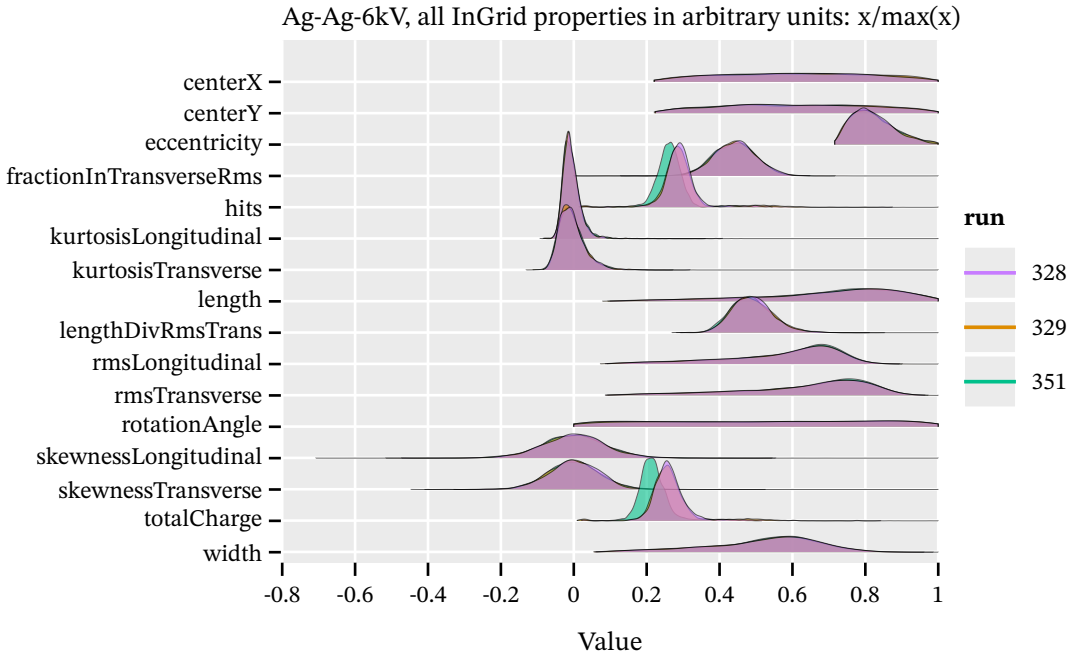


Figure H.5.: Ridgeline plot of kernel density estimations of all cluster properties split by each CDL run. Target/filter: Ag-Ag-6kV

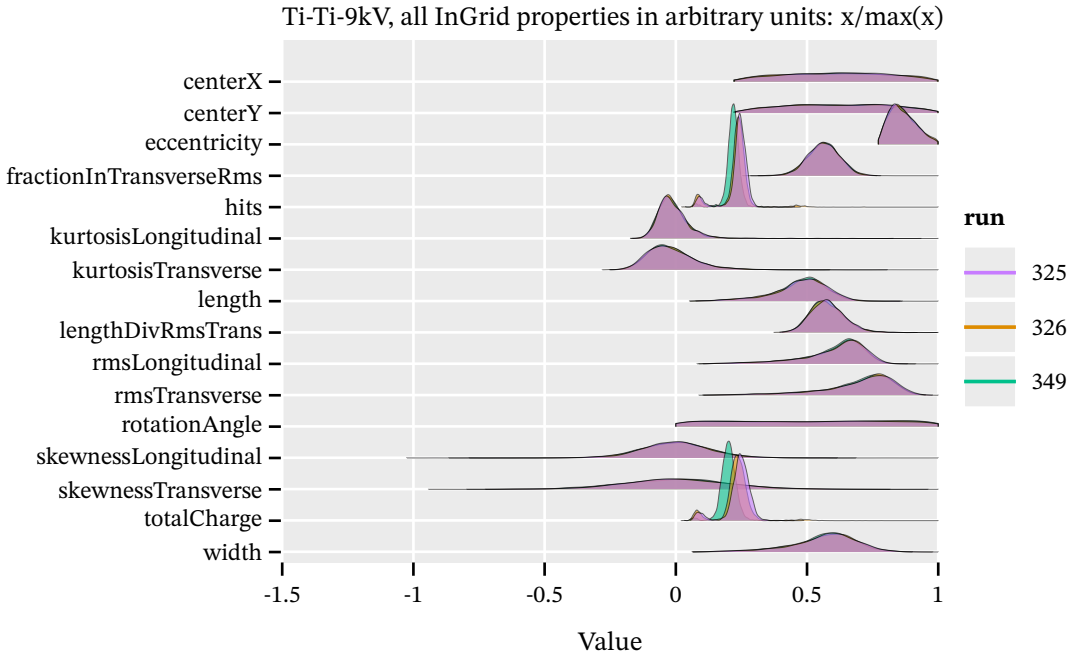


Figure H.6.: Ridgeline plot of kernel density estimations of all cluster properties split by each CDL run. Target/filter: Ti-Ti-9kV

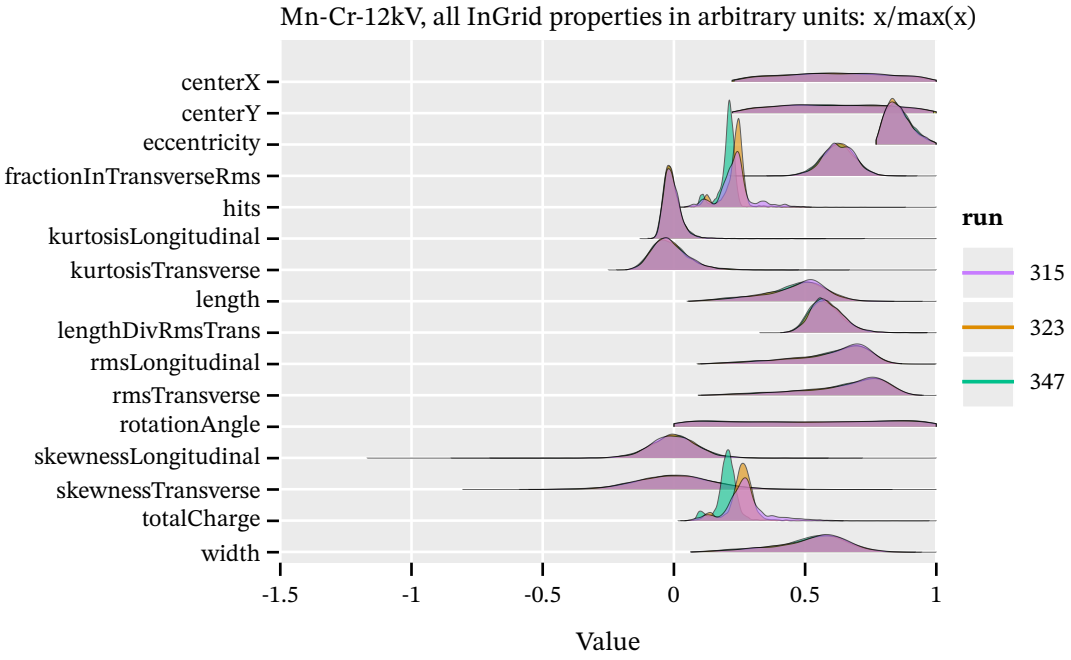


Figure H.7.: Ridgeline plot of kernel density estimations of all cluster properties split by each CDL run. Target/filter: Mn-Cr-12kV

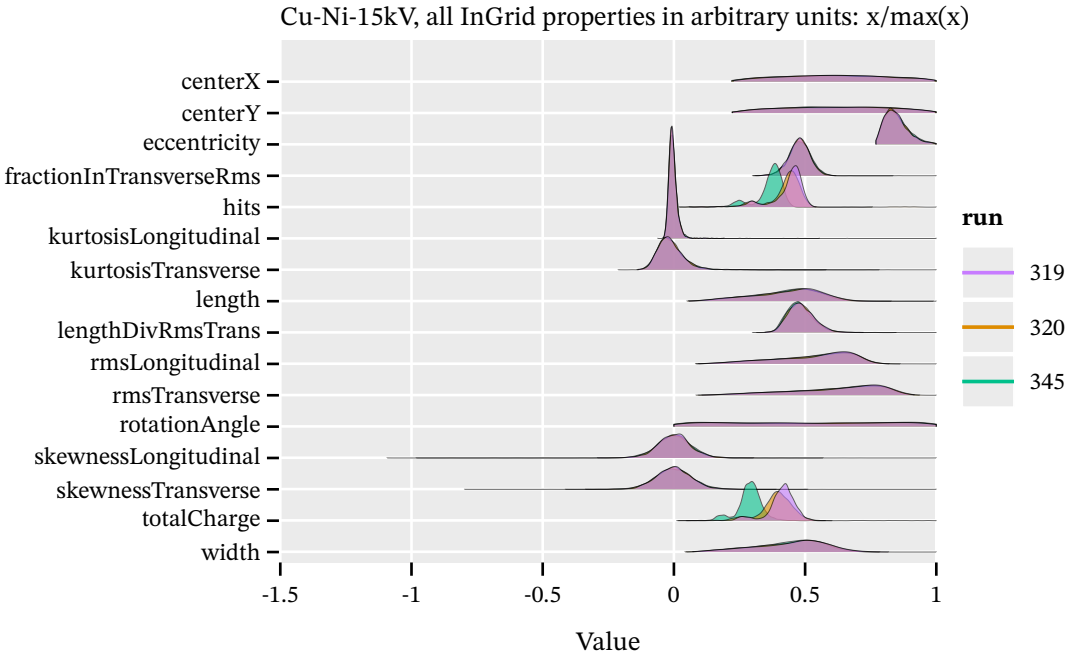


Figure H.8.: Ridgeline plot of kernel density estimations of all cluster properties split by each CDL run. Target/filter: Cu-Ni-15kV

I. Morphing of CDL reference spectra Appendix

This appendix contains further figures about the interpolation of the reference distributions entering the likelihood method as discussed in sec. 11.2.6 and sec. 11.2.7. Fig. I.1 directly shows the impact of interpolating the reference distributions on the likelihood values (colored points for each different target/filter combination) and the cut value (black line). Both the points and the cut values become smoother utilizing the interpolation. Note that even the interpolation is not perfectly smooth, as the underlying reference data is still binned to a histogram.

Further, section I.1 contains tilemaps of the not interpolated reference data for each dataset and sec. I.2 shows the fully interpolated space for each dataset. Finally, sec. I.3 contains the histograms of the binwise linear interpolations where for each ridge one dataset is skipped.

See the extended thesis for the development notes of the interpolation technique, including different approaches (e.g. binwise spline interpolation instead of linear interpolation).

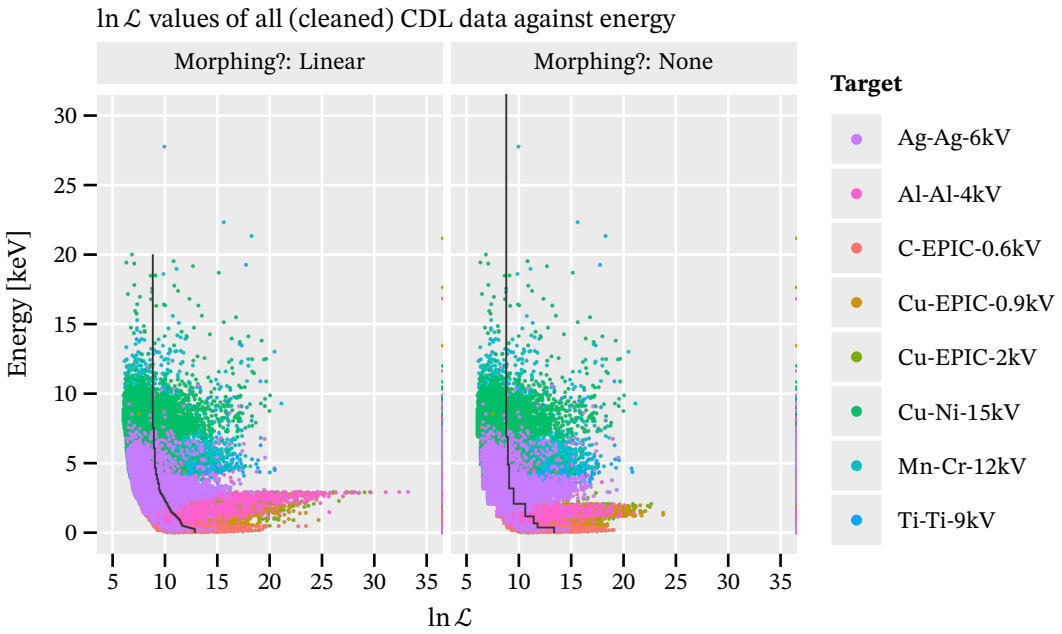


Figure I.1.: ln ℒ values for all the cleaned CDL cluster data against the energy of the cluster. Left is the binwise linear interpolation and right is the calculation using the old [134] approach of fixed energy intervals. The bin wise linear interpolation helps to provide a smoother description of the ln ℒ data.

I.1. Tilemap of each likelihood dataset

The tilemaps shown here should be compared to the resulting raster plots of the next section, sec. I.2. For each dataset entering the likelihood method the reference distribution computed for each target/filter combination is spread over its applicable energy range. In the plots the colormap is the height of the distribution. The fluorescence lines are indicated that define the X-ray energy of each target. This illustrates how a cluster energy is mapped to a probability from each of the 8 target/filter distributions. The distinct jump in the distributions is visible in the form of a cut along the horizontal axis at different energies.

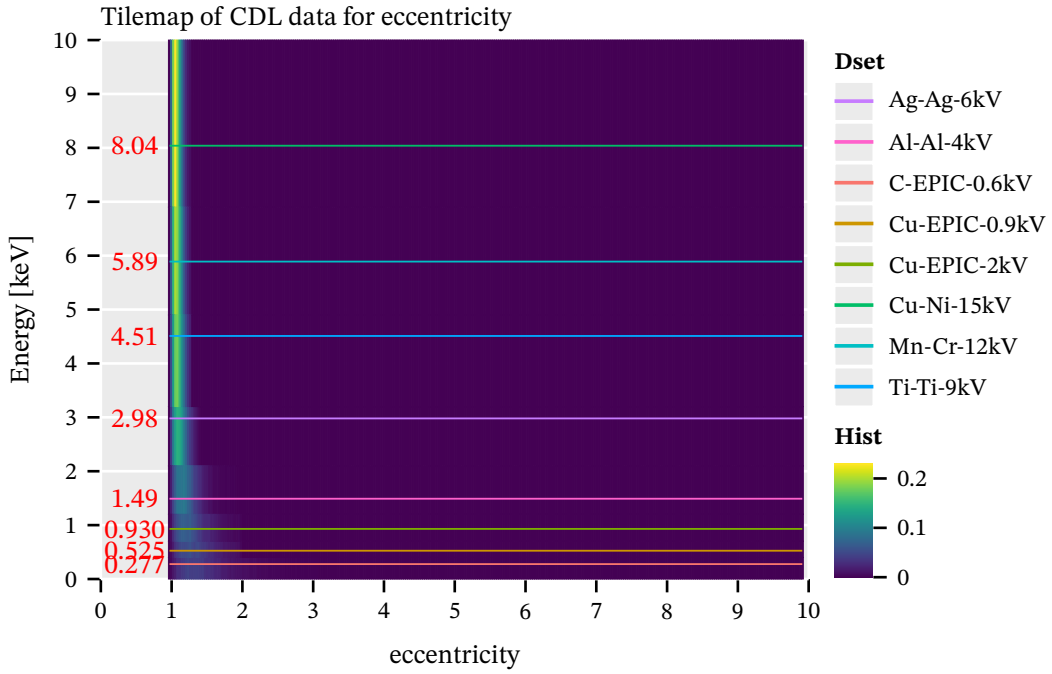


Figure I.2.: Tilemap of the eccentricity dataset against energy. The colormap corresponds to the height of the eccentricity distribution at that point. Along the energy axis the data is constant within the applicable energy range of the target/filter kind. The energy of each fluorescence line is indicated.

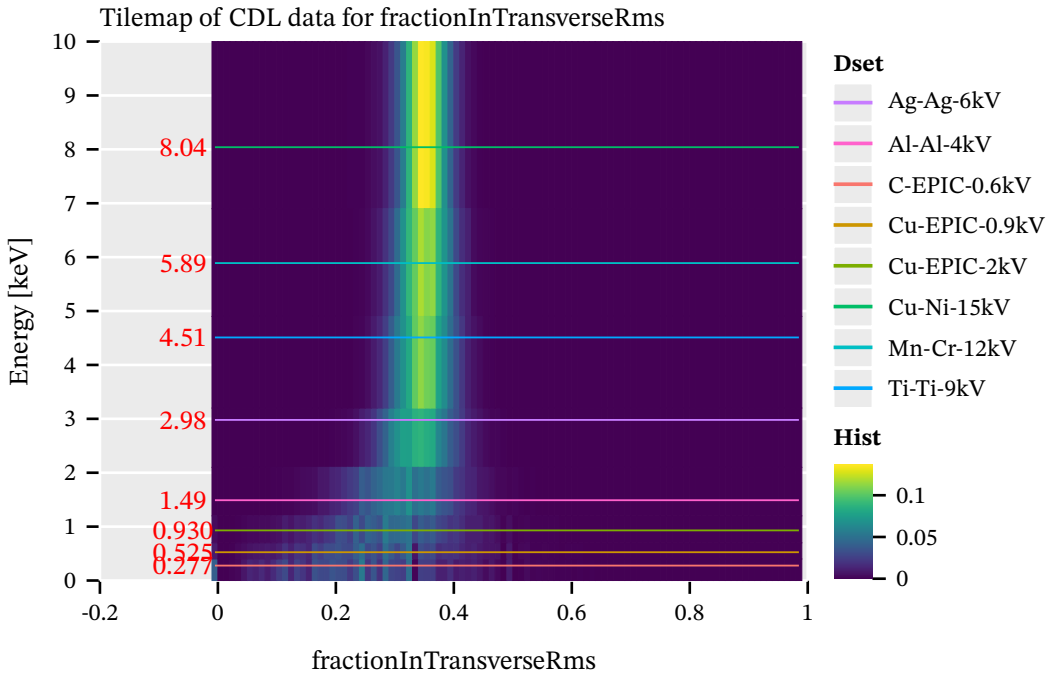


Figure I.3.: Tilemap of the fractionInTransverseRms dataset against energy. The colormap corresponds to the height of the fractionInTransverseRms distribution at that point. Along the energy axis the data is constant within the applicable energy range of the target/filter kind. The energy of each fluorescence line is indicated.

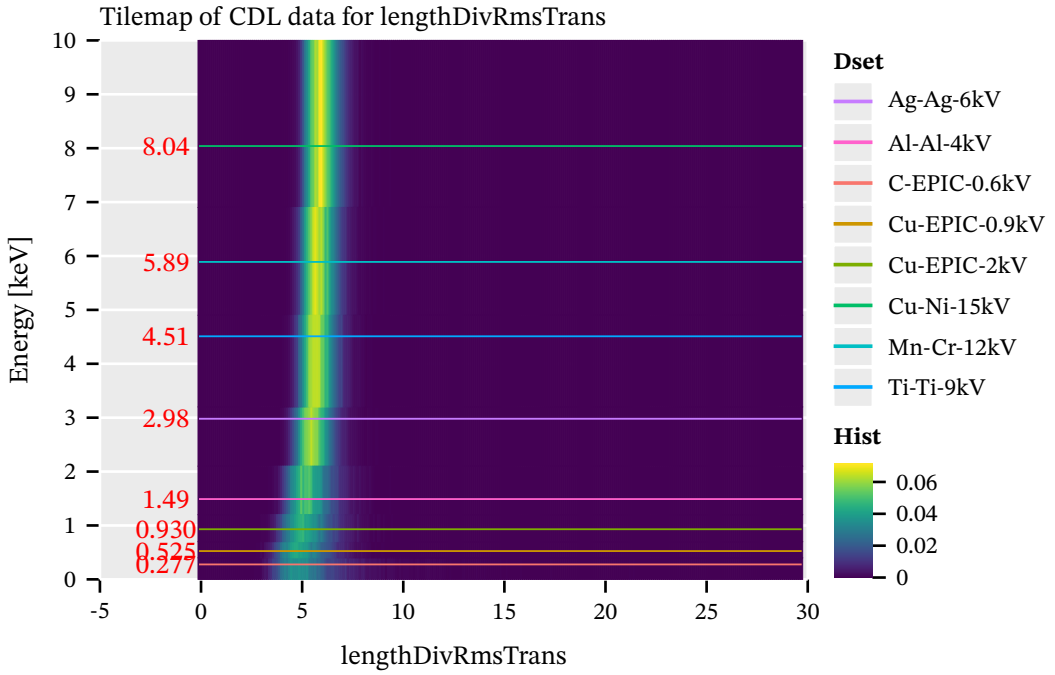


Figure I.4.: Tilemap of the lengthDivRmsTrans dataset against energy. The colormap corresponds to the height of the lengthDivRmsTrans distribution at that point. Along the energy axis the data is constant within the applicable energy range of the target/filter kind. The energy of each fluorescence line is indicated.

I.2. Interpolation of each likelihood dataset

In contrast to the plots of the previous section, sec. I.1 all plots of the different datasets entering the likelihood method here show the distribution in property / energy space after binwise linear interpolation. Compared to the previous plots it results in a smooth transition over the entire energy range.

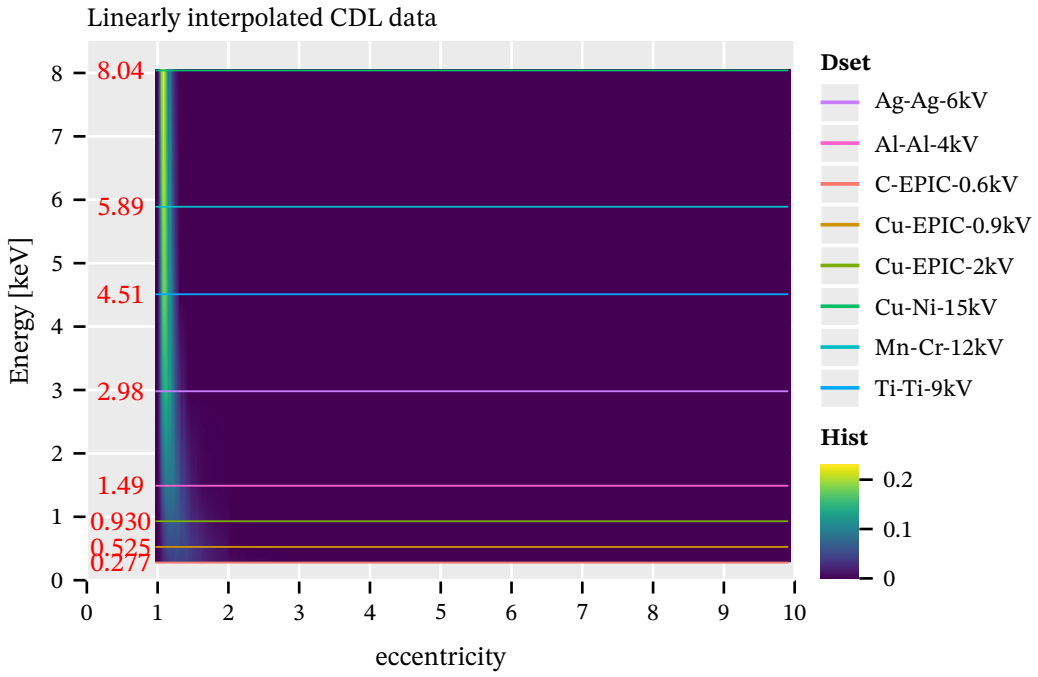


Figure I.5.: Raster plot of the eccentricity dataset against energy after binwise linear interpolation. The colormap corresponds to the height of the eccentricity distribution at that point. The energy of each fluorescence line is indicated.

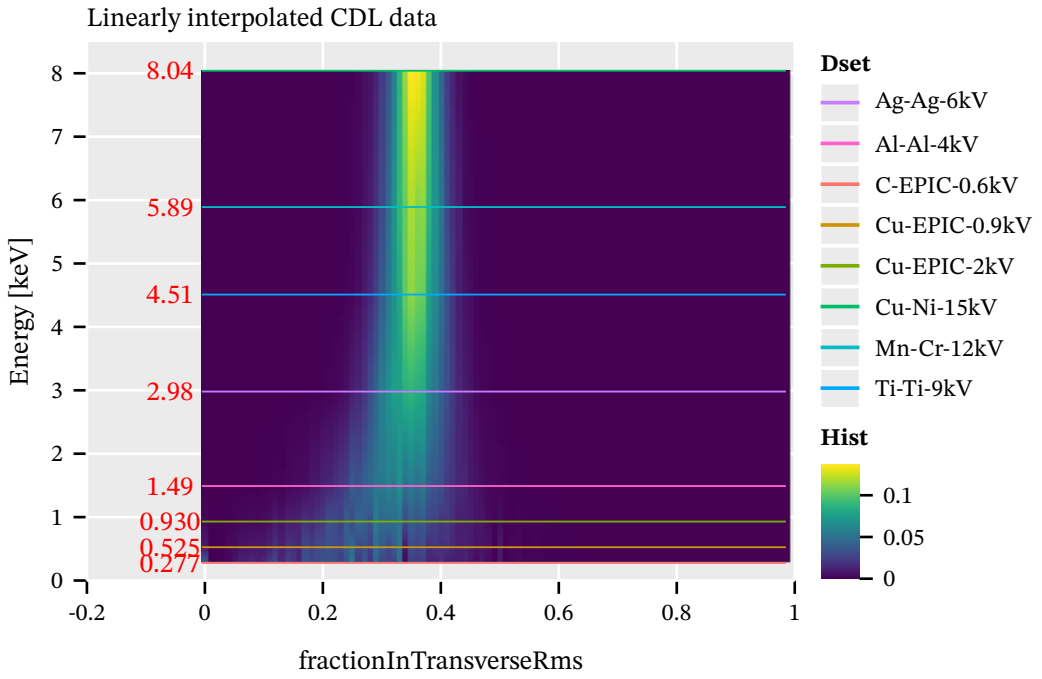


Figure I.6.: Raster plot of the fractionInTransverseRms dataset against energy after binwise linear interpolation. The colormap corresponds to the height of the fractionInTransverseRms distribution at that point. The energy of each fluorescence line is indicated.

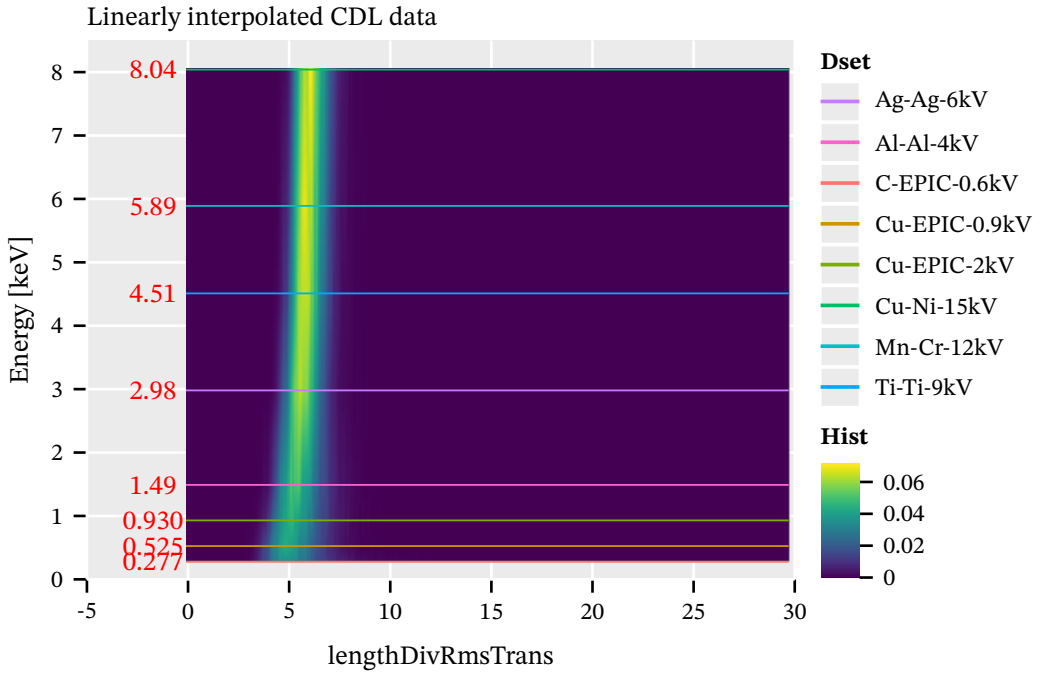


Figure I.7.: Raster plot of the lengthDivRmsTrans dataset against energy after binwise linear interpolation. The colormap corresponds to the height of the lengthDivRmsTrans distribution at that point. The energy of each fluorescence line is indicated.

I.3. Binwise linear interpolations for each likelihood dataset

The figures in this section show the binwise linear interpolation by skipping the dataset in each ridge and interpolating using the nearest neighbors above/below in energy. These are the same as shown in fig. 11.8(a) for all three datasets (i.e. fig. I.9 is the same). As mentioned in the main text, skipping the dataset at each ridge means the interpolation errors are much larger than used in practice (as shown in the previous section I.2).

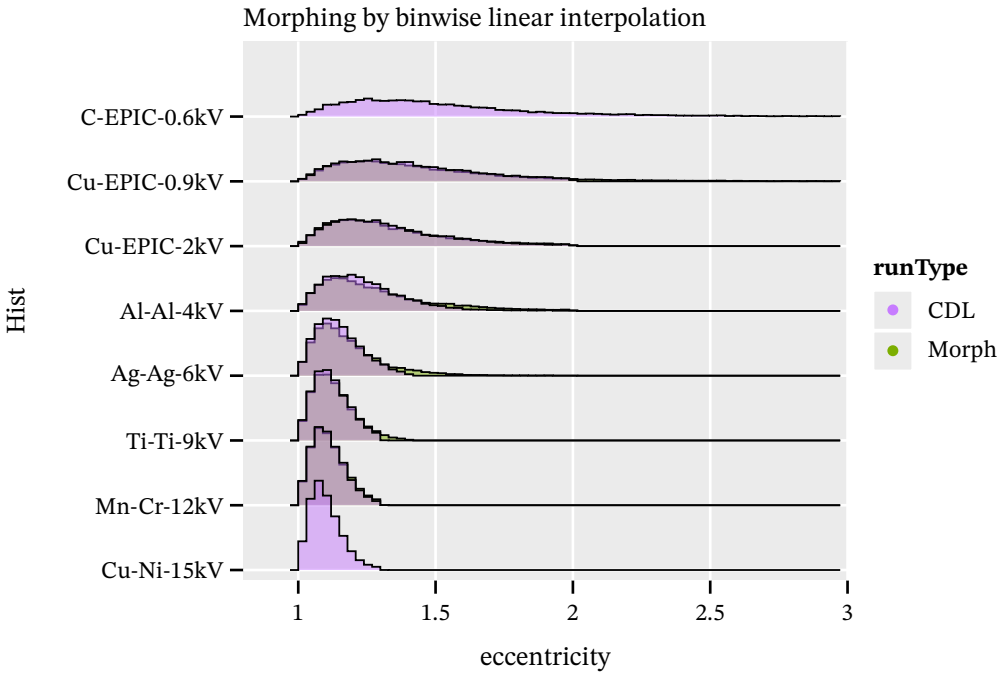


Figure I.8.: Binwise linear interpolation of the eccentricity dataset skipping the target/filter dataset being interpolated.

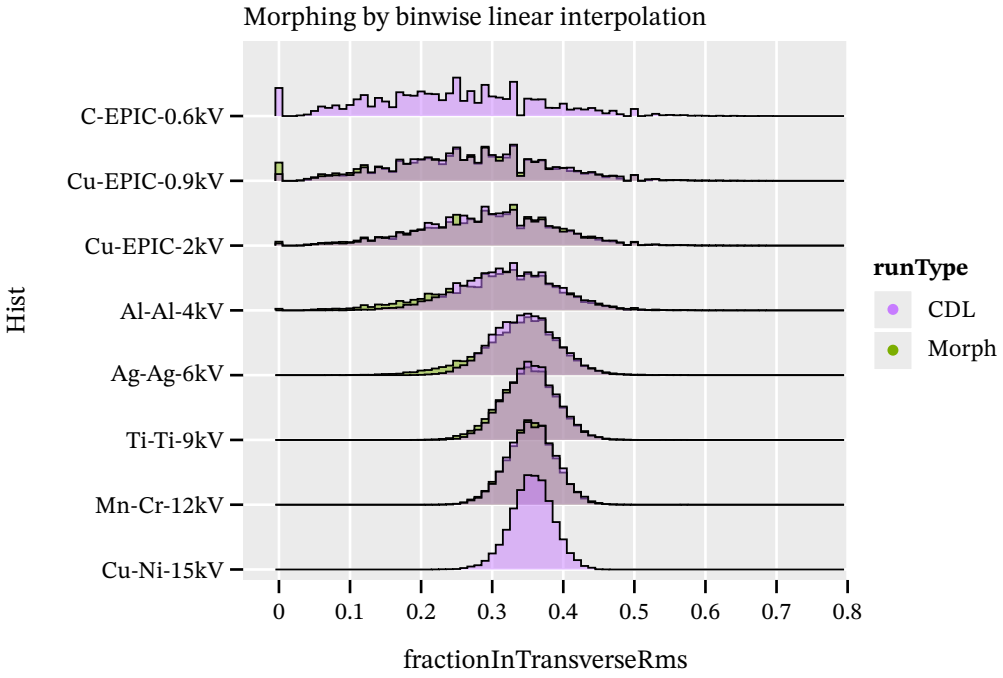


Figure I.9.: Binwise linear interpolation of the fractionInTransverseRms dataset skipping the target/filter dataset being interpolated.

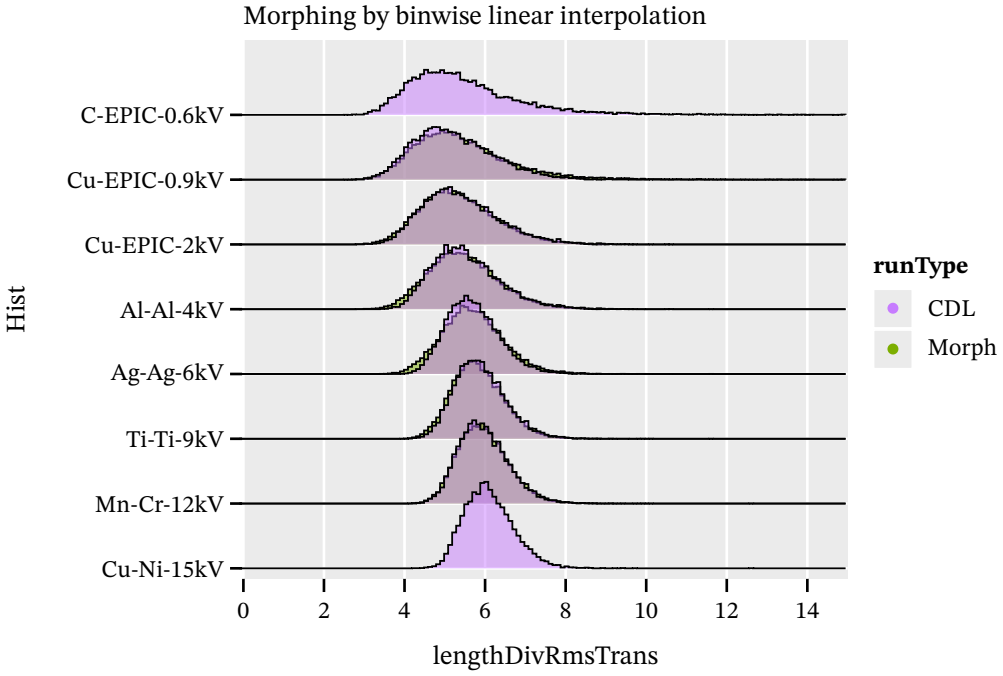


Figure I.10.: Binwise linear interpolation of the lengthDivRmsTrans dataset skipping the target/filter dataset being interpolated.

J. Occupancy maps Appendix

In this appendix we find a few plots of the entire Septemboard activity during the CAST Run-2 and Run-3 campaigns. For each a plot of the raw pixel activity in the form of counts (how many times was each pixel activated) and in the form of summing the Time over Threshold (ToT) values is shown. In both cases the color scale is cut to the 95^{th} percentile of the data for better visibility. In each plot small regions that see infrequent sparks show up. Different chips have different amounts of activity, due to imperfect calibrations, in particular of the 6 outer chips.

J.1. Run-2

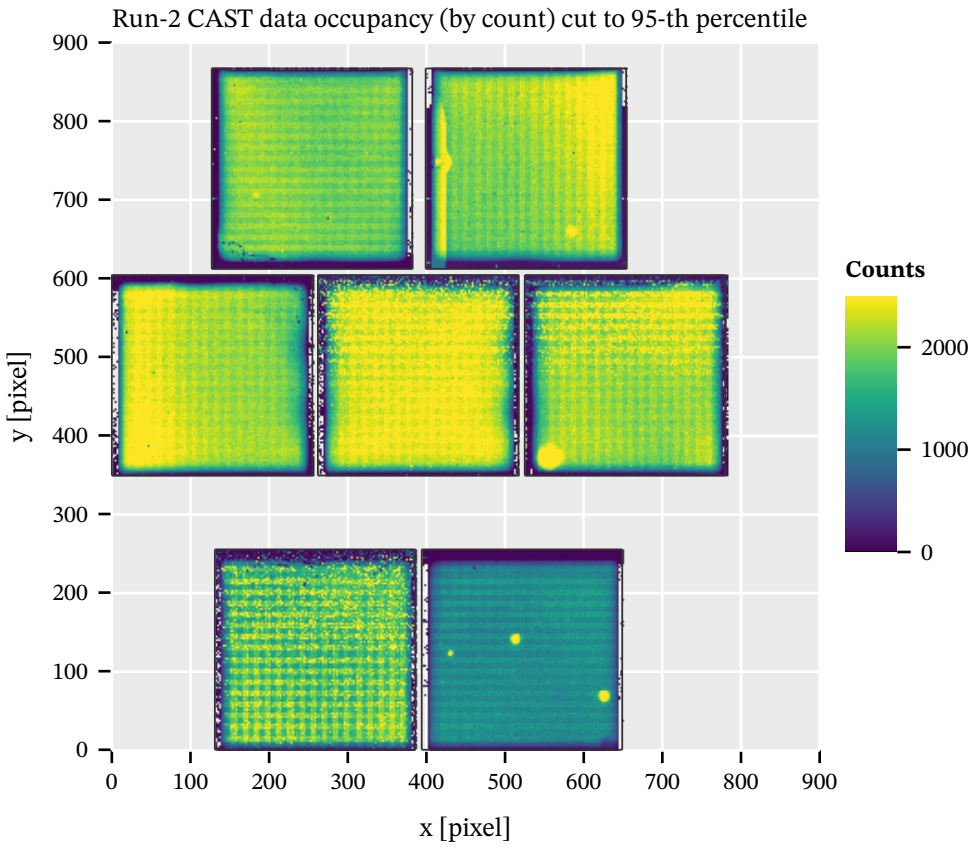


Figure J.1.: Occupancy maps of the Septemboard during the Run-2 data taking campaign at CAST based on the raw pixel activity counts. Color scale saturates at 95$^{\text{th}}$ percentile of all data. Some regions with sparks are visible. Top right chip on the left edge and bottom right, center right chip bottom left and bottom right chip in multiple spots. Generally activity is slightly different between chips, due to different threshold calibrations.

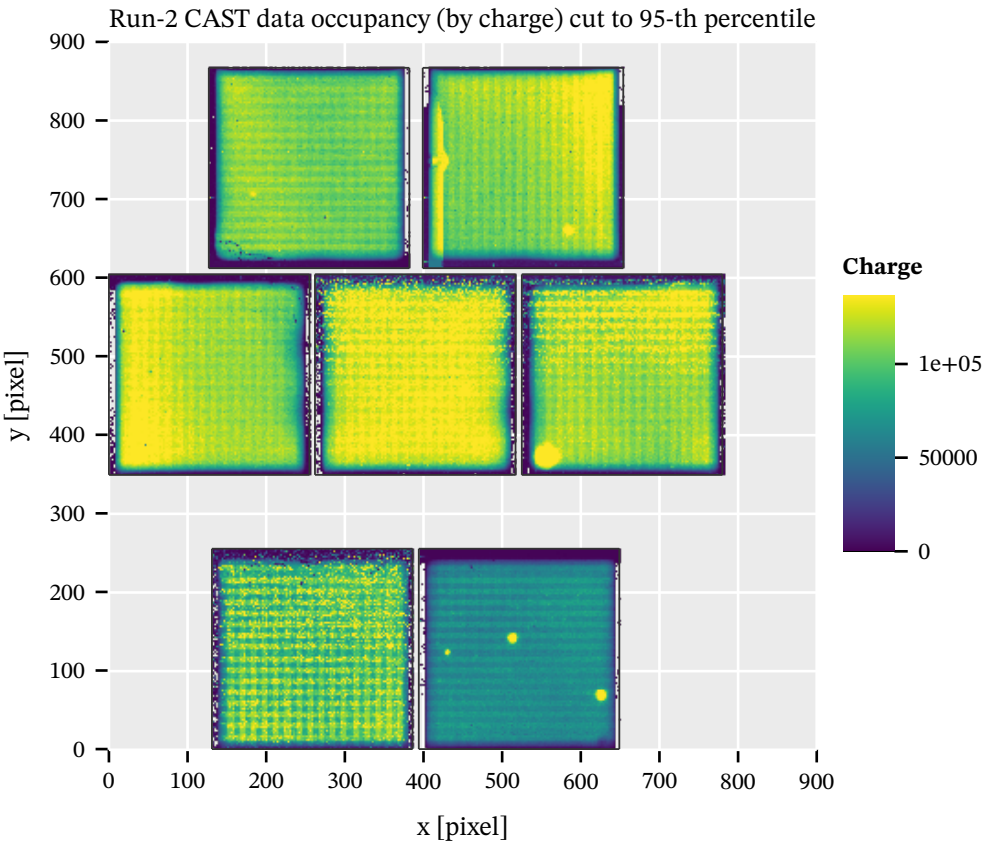


Figure J.2.: Occupancy maps of the Septemboard during the Run-2 data taking campaign at CAST based on the sum of raw pixel ToT values. Color scale saturates at 95$^{\text{th}}$ percentile of all data. Any pixel activity exceeding $\texttt{ToT} > 1000$ are filtered out.

J.2. Run-3

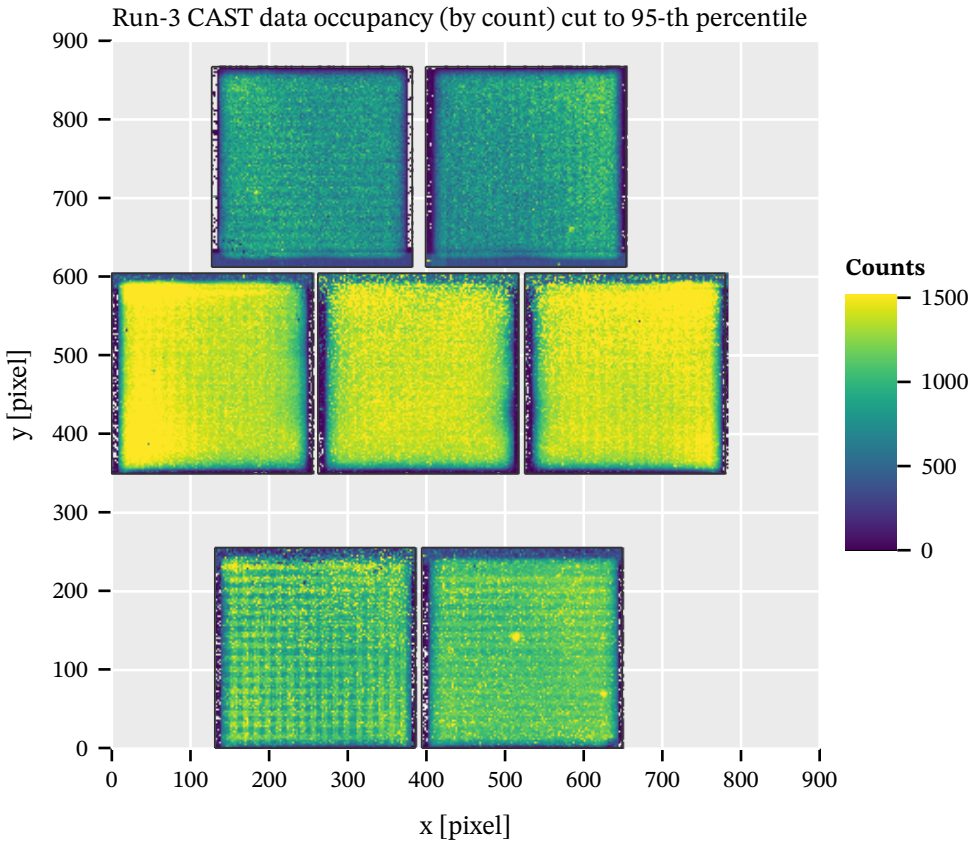


Figure J.3.: Occupancy maps of the Septemboard during the Run-3 data taking campaign at CAST based on the raw pixel activity counts. Color scale saturates at 95$^{\text{th}}$ percentile of all data. Less spark activity visible than in Run-2. Differing levels of activity are still visible.

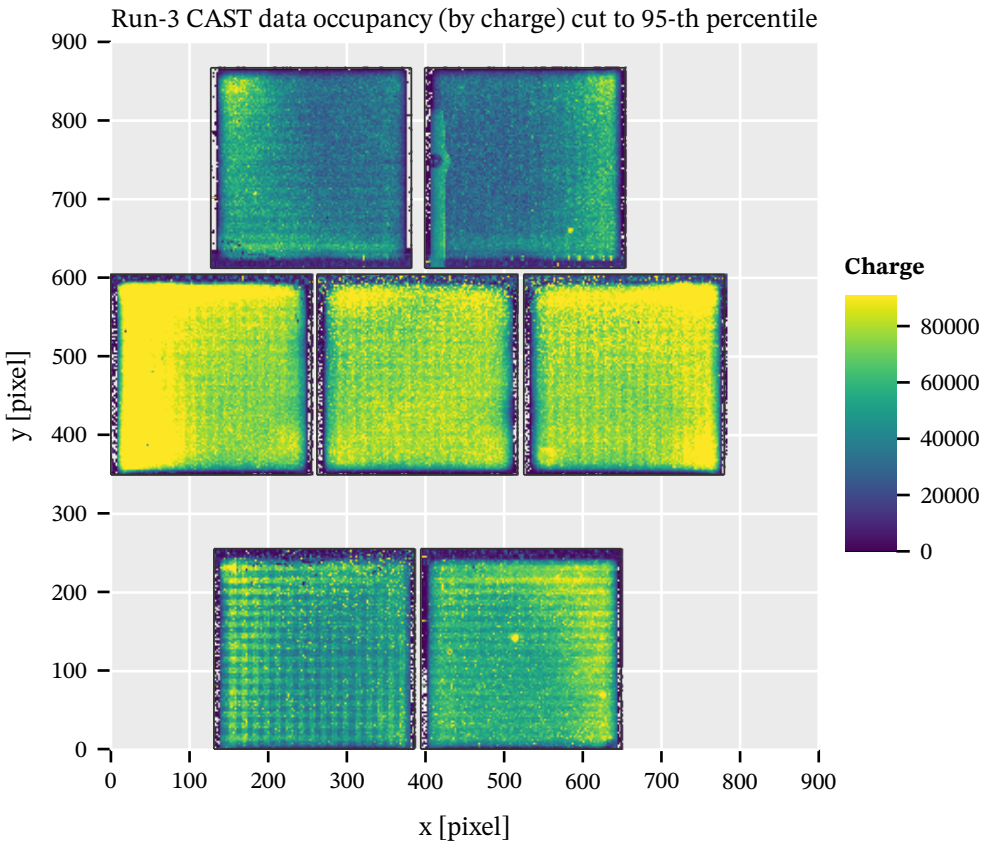


Figure J.4.: Occupancy maps of the Septemboard during the Run-3 data taking campaign at CAST based on the sum of raw pixel ToT values. Color scale saturates at 95^{th} percentile of all data. Any pixel activity exceeding $\texttt{ToT} > 1000$ are filtered out.

K. FADC Appendix

This appendix contains a few extra figures about the FADC signals. Sec. K.1 contains additional figures – similar to fig. 11.20 used in the main thesis – about the distribution of rise and fall times. Sec. K.2 contains more specifics related to the FADC veto. In the context of the FADC veto in the main body the expected cluster size is mentioned, which is supported by a figure in sec. K.3.

K.1. FADC rise and fall time

These figures show a few extra figures of the FADC signal rise and fall times comparing the ^{55}Fe calibration data with background data, for Run-2 and Run-3 of the CAST data taking campaign. In the Run-2 figures more variation is visible, due to the different FADC amplifier settings. These are similar (and include) fig. 11.20 shown in the main body. See in particular figures K.5 and K.6 for the impact of the rise and fall time based on the different settings. The fall time figures show that the difference between signal and background is not very pronounced. This is expected, as the fall times are dominated by the resistor-capacitor behavior of the circuit.

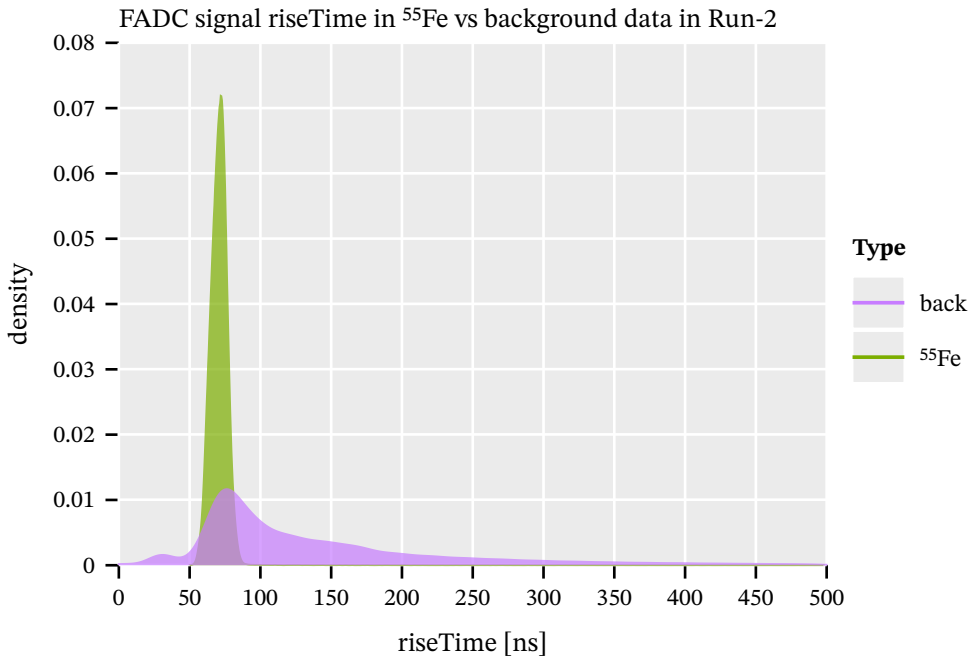


Figure K.1.: KDE of the rise time of the FADC signals in the ^{55}Fe and background data of the CAST Run-2 dataset. The X-ray data is a single peak with a mean of about 70 ns while the background distribution is extremely wide, motivating a veto based on this data.

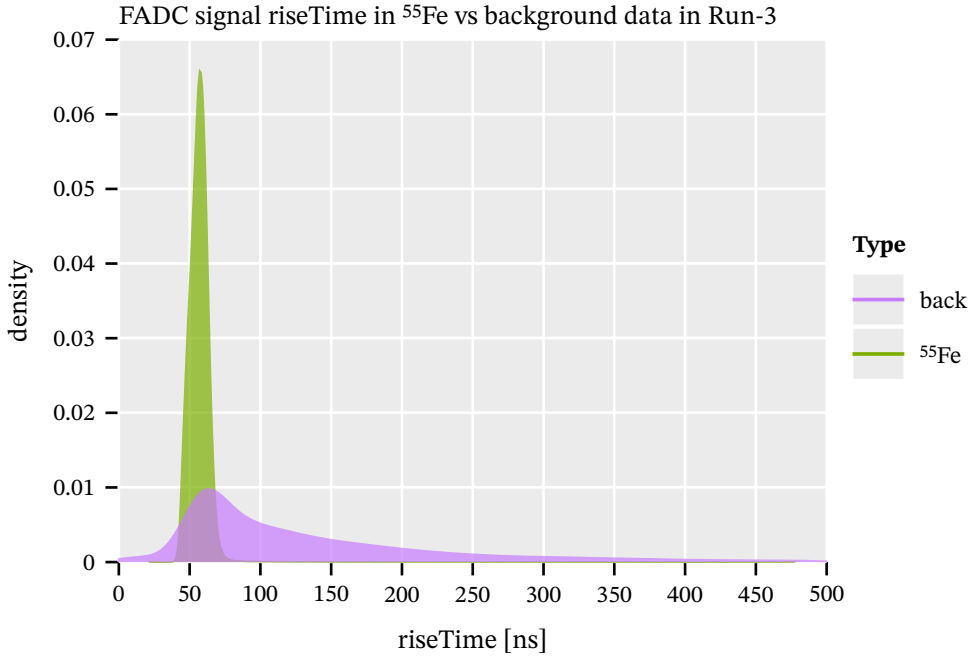


Figure K.2.: KDE of the rise time of the FADC signals in the ^{55}Fe and background data of the CAST Run-3 dataset. The X-ray data is a single peak with a mean of about 55 ns while the background distribution is extremely wide, motivating a veto based on this data.

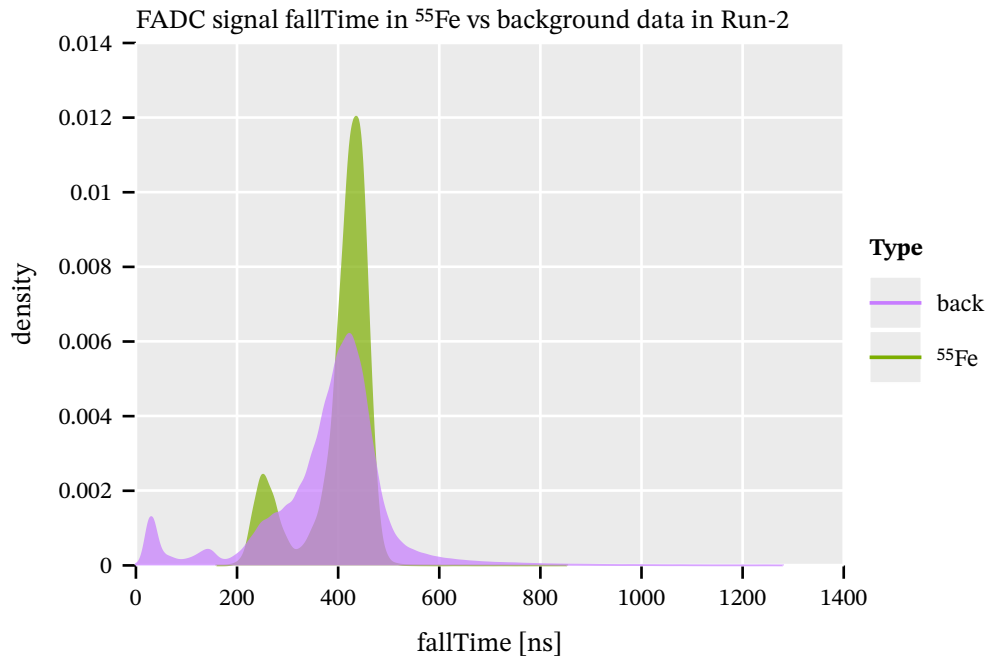


Figure K.3.: KDE of the rise time of the FADC signals in the ^{55}Fe and background data of the CAST Run-2 dataset. Difference in the two signal types is negligible. Multi-peak structure is due to different FADC settings, see fig. K.5.

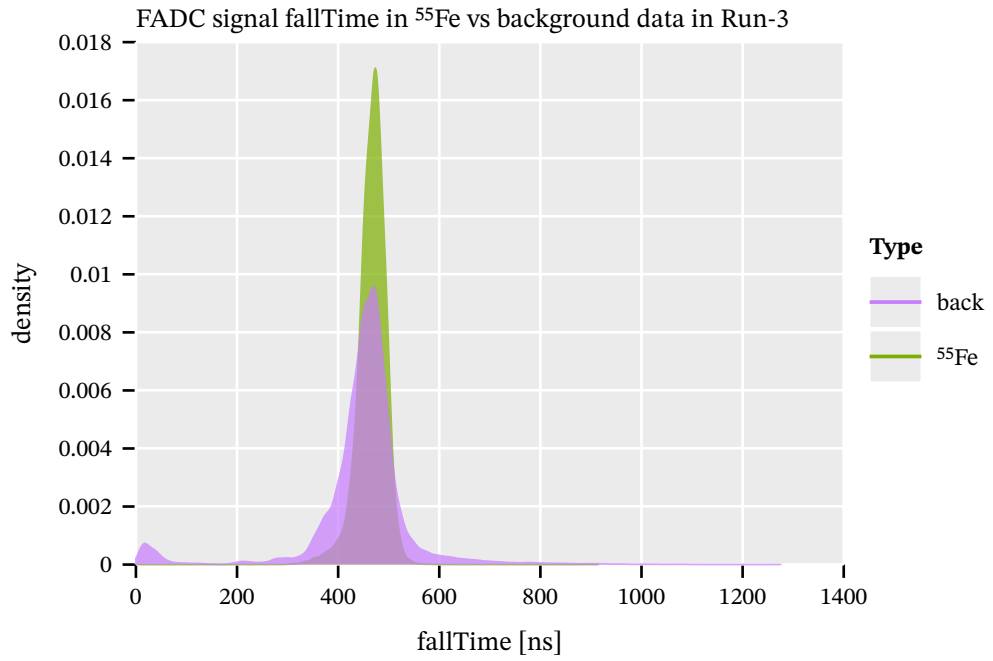


Figure K.4.: KDE of the fall time of the FADC signals in the ^{55}Fe and background data of the CAST Run-3 dataset. Difference in the two signal types is negligible.

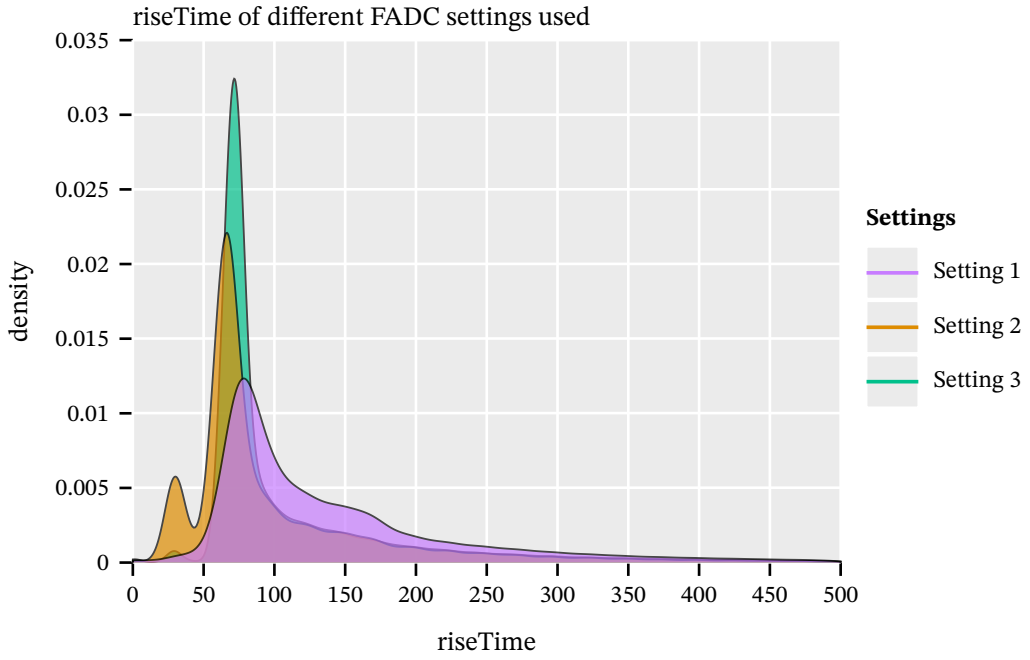


Figure K.5.: KDE of the rise time of the FADC signals in the ^{55}Fe data of Run-2 split by the different FADC settings.

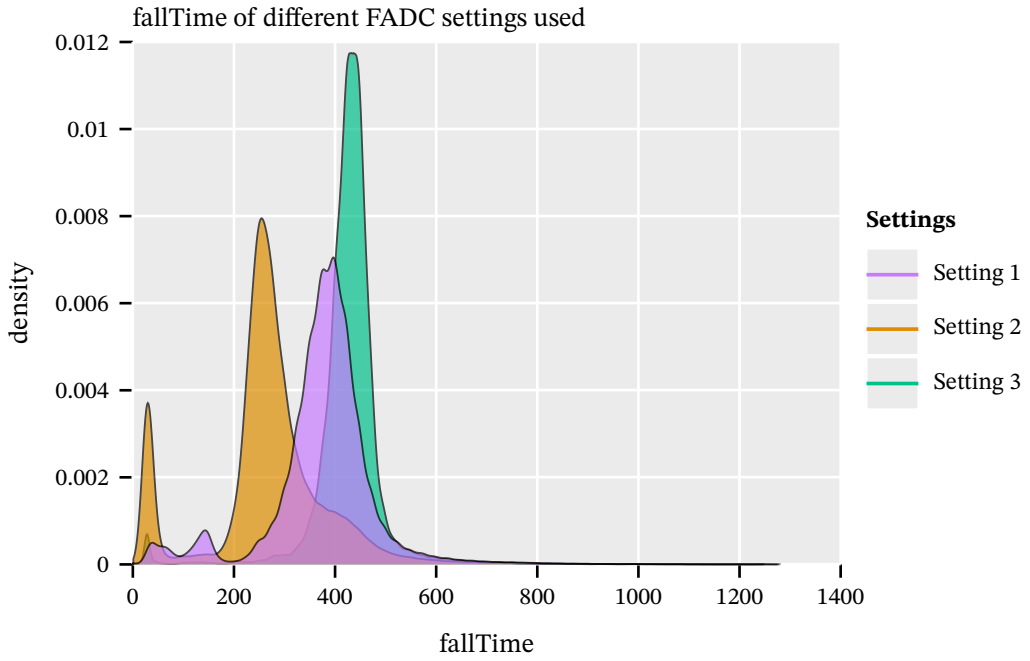


Figure K.6.: KDE of the fall time of the FADC signals in the ^{55}Fe data of Run-2 split by the different FADC settings. Impact on the fall time is more pronounced than on the rise time above.

K.2. FADC veto

Only briefly mentioned in sec. 11.5.2 is the addition of a cut on the skewness of the FADC signal, namely a skewness < -0.4. Fig. K.7 shows the FADC signal rise times of all events in Run-2 and fig. K.8 for Run-3 against the skewness of the signal (i.e. a measure of how one sided the signal is due to the negative from baseline shape of a single FADC pulse). The color scale indicates whether the event is considered 'noisy' by our noise filter.

We can see the majority of all noise events above skewness values of ≥ 0.4, roughly. There are pockets of non-noisy events above this skewness value, in particular rise times of 180 to 210 and skewness -0.7 to 0.3 in Run-2. Events in this range were inspected by eye (see extended thesis). They contain other noisy events not registered by the noise filter and certain types of multi cluster events resulting in peculiar shapes.

Finally, fig. K.9 shows the achievable signal efficiencies against background suppression as a function of the rise time. In this case only the upper range is cut (in practice the cut is symmetric on lower and upper end). This illustrates the efficiency of the FADC veto without any additional detector parts. At our used 99 % upper range cut the background is suppressed to about 60 % of the full background in the energy range of the ^{55}Fe photopeak.

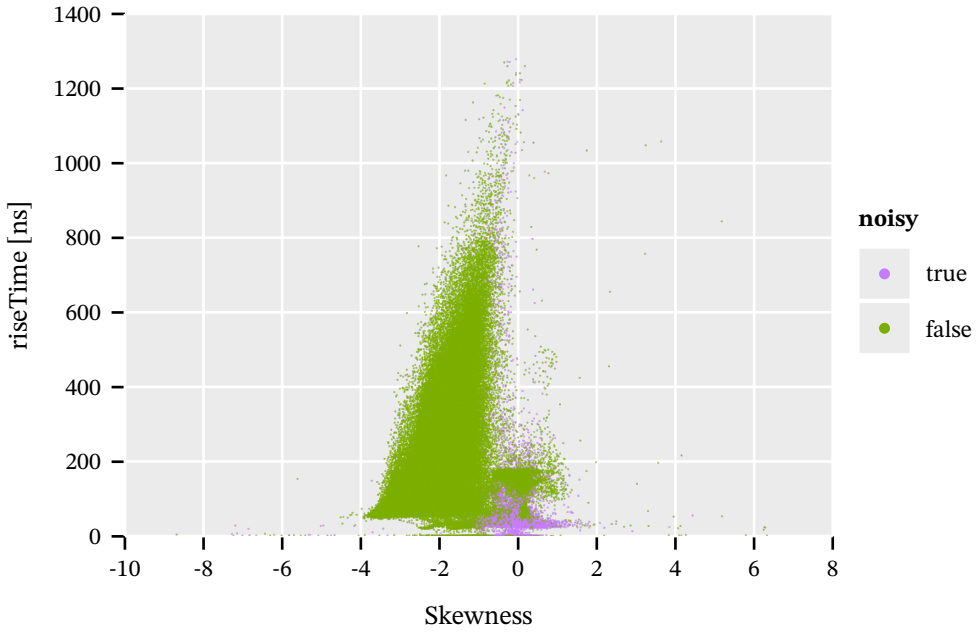


Figure K.7.: Scatter plot of all FADC events in Run-2 of the rise time of each event against the skewness. Colorcoded is whether each event is considered noisy by our noise filter.

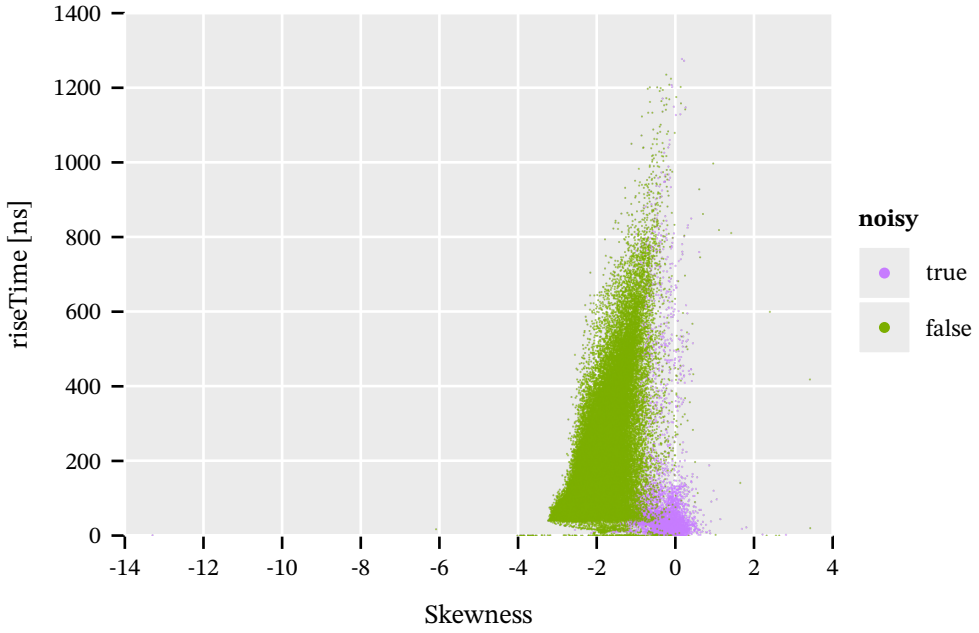


Figure K.8.: Scatter plot of all FADC events in Run-3 of the rise time of each event against the skewness. Colorcoded is whether each event is considered noisy by our noise filter.

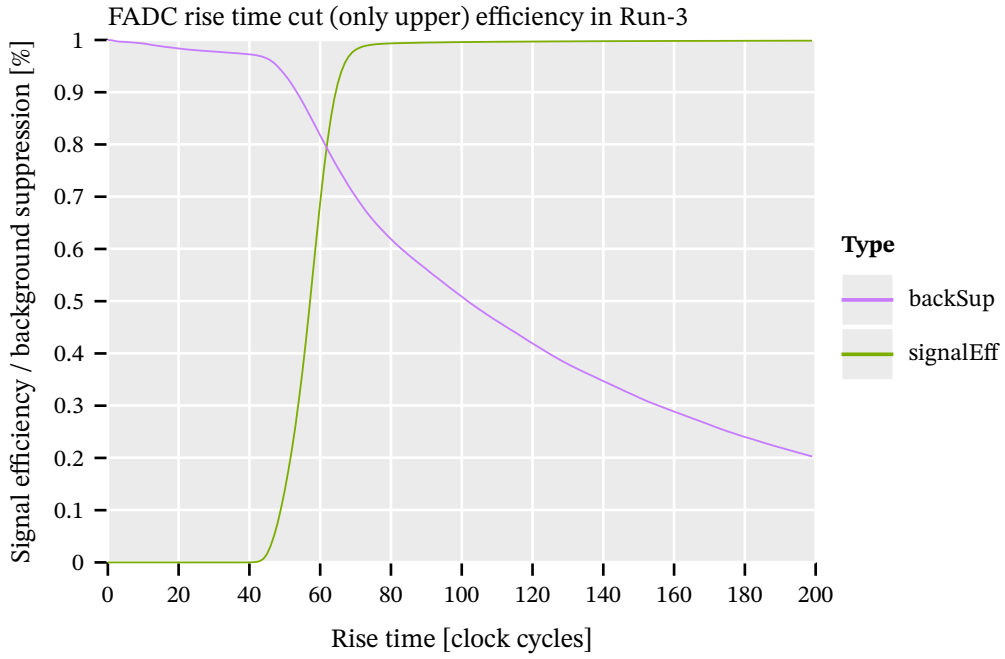


Figure K.9.: Signal efficency achievable using only the FADC veto for ^{55}Fe calibration data by cutting on the rise time (only the upper end) compared to the corresponding background suppression (for Run-3 data).

K.3. Expected cluster size

While not strictly speaking FADC data, the expected size of clusters was mentioned in sec. 11.5.2 to be about 6 mm in length for ^{55}Fe calibration data. This is shown in fig. K.10.

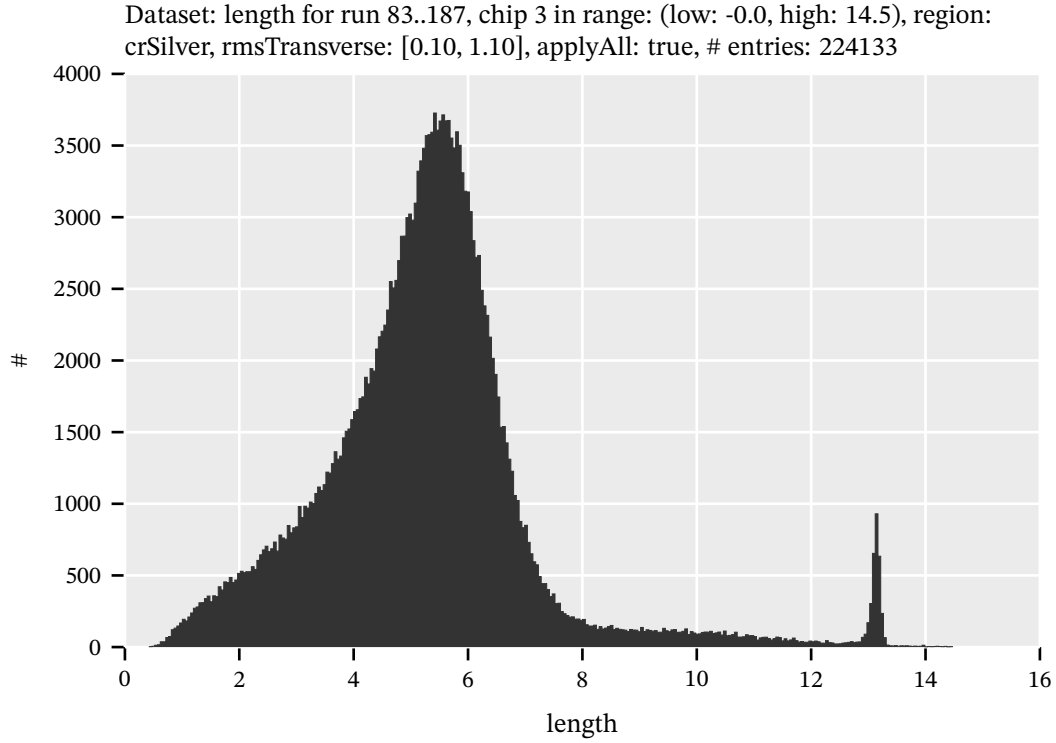


Figure K.10.: Cluster lengths in millimeter of ^{55}Fe calibration data during the Run-2 data taking campaign. These 5.9 keV clusters peak at around 5.5 mm in length with significant drop in statistics from 6 mm.

L. Raw data and background rates Appendix

In this appendix we find a few more figures and numbers about the background rates and a comparison to the raw data rates. Fig. L.1 shows the raw data rate of the full center chip of the entire CAST data taking campaign in the same units as used for the normal background rate plots. The background increases towards low energies, but a hump is visible in the 2 to 3 keV range, matching the expected energies for muon tracks. The rates are roughly four orders of magnitude larger than the best achieved background rates.

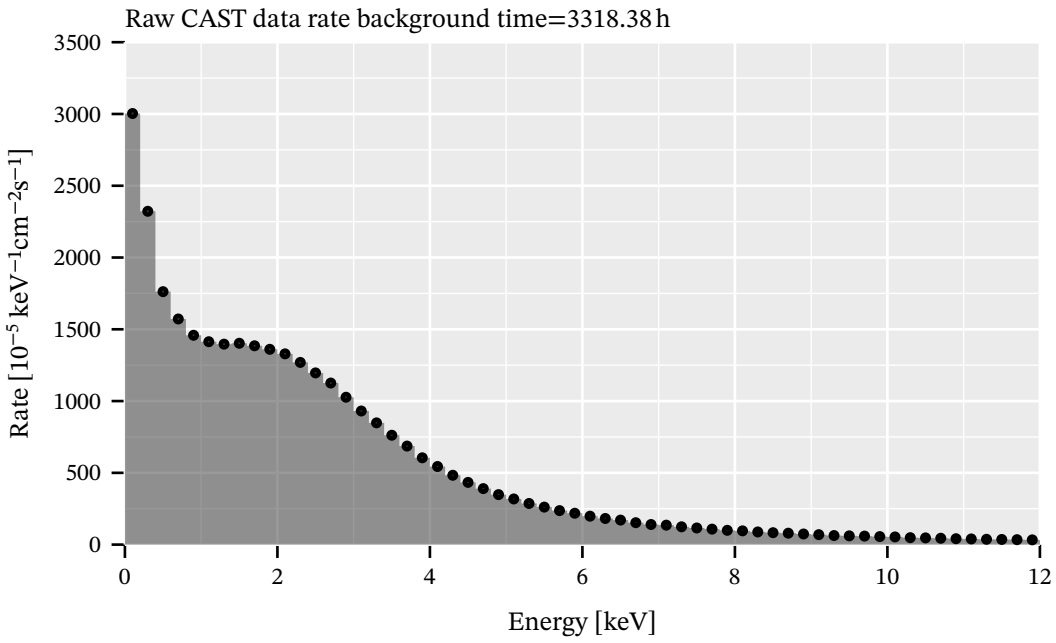


Figure L.1.: All CAST data of the entire center chip without any classifier or other cuts. Uses the same units as all background plots for easier comparison.

Further, in sec. L.1 we find background rates not only for the $5 \times 5\,\text{mm}^2$ region, but for the entire center chip. Both to compare with fig. L.1 as well as to get a better idea of the efficiency of the vetoes.

L.1. Background rates over full chip

Fig. L.2 shows the effect of the different vetoes over the entire chip using the $\ln \mathcal{L}$ method at 80 % software efficiency as a base. It better highlights the efficiency of some vetoes, for example the FADC

veto is more effective over the entire chip, than in the center $5 \times 5\,\mathrm{mm}^2$ region alone (compare with fig. 11.21). Of course, the septem and line vetoes are even more efficient over the entire chip, as expected and previously seen in the form of the background suppression plots (for example fig. 11.29).

Fig. L.3 shows how the different software efficiencies affect the background rate comparing different $\ln \mathcal{L}$ and MLP efficiencies including all vetoes.

Finally, tab. L.1 shows the mean background rates over the entire chip for all considered $\ln \mathcal{L}$ and MLP veto setups in the energy range 0.2 to 8 keV. It is the equivalent of tab. 11.8 for the full chip.

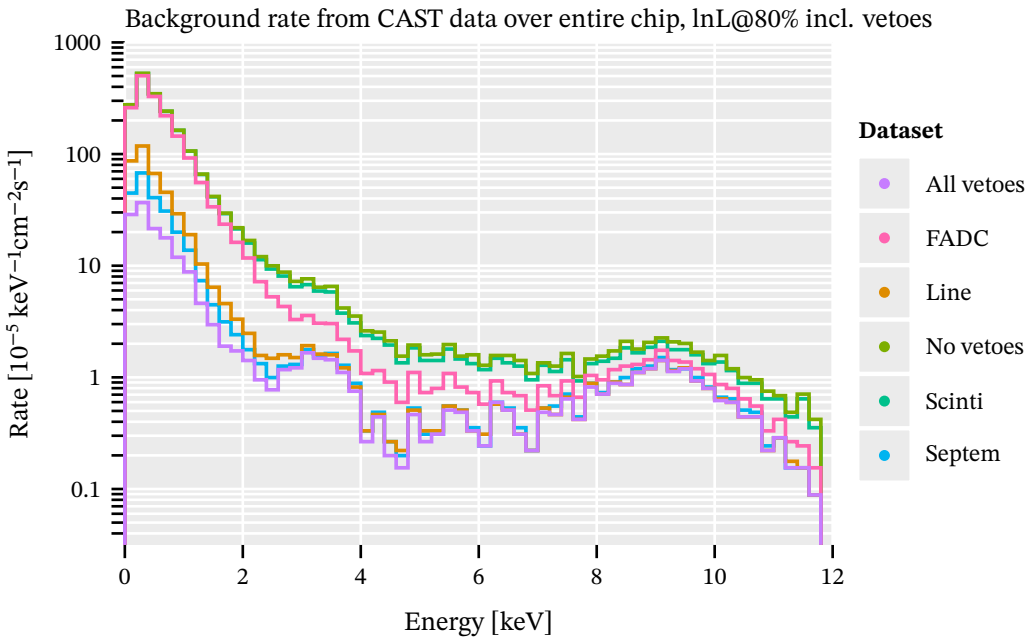


Figure L.2.: Background rate over the entire center chip of all CAST data, comparing the $\ln \mathcal{L}$ cut method at 80 % software efficiency without any vetoes with each veto setup (vetoes are additive again).

Classifier	ε_{eff}	Scinti	FADC	Septem	Line	$\varepsilon_{\text{total}}$	Rate [$\text{keV}^{-1}\,\text{cm}^{-2}\,\text{s}^{-1}$]
MLP	0.865	true	true	true	true	0.621	$1.736(31) \times 10^{-5}$
MLP	0.912	true	true	true	true	0.655	$2.264(36) \times 10^{-5}$
LnL	0.700	true	true	true	true	0.503	$2.546(38) \times 10^{-5}$
LnL	0.800	true	true	true	true	0.574	$3.301(43) \times 10^{-5}$
MLP	0.957	true	true	true	true	0.687	$3.555(45) \times 10^{-5}$
LnL	0.900	true	true	true	true	0.646	$4.513(51) \times 10^{-5}$
MLP	0.865	true	true	false	true	0.729	$4.900(53) \times 10^{-5}$
LnL	0.800	true	true	true	false	0.615	$5.482(56) \times 10^{-5}$
MLP	0.983	true	true	true	true	0.706	$5.792(57) \times 10^{-5}$
MLP	0.912	true	true	false	true	0.769	$6.258(60) \times 10^{-5}$
LnL	0.700	true	true	false	true	0.590	$6.595(61) \times 10^{-5}$
LnL	0.800	true	true	false	true	0.674	$8.429(69) \times 10^{-5}$
MLP	0.957	true	true	false	true	0.807	$9.120(72) \times 10^{-5}$
LnL	0.900	true	true	false	true	0.759	$1.1070(79) \times 10^{-4}$
MLP	0.983	true	true	false	true	0.829	$1.3448(87) \times 10^{-4}$
MLP	0.865	false	false	false	false	0.865	$3.099(13) \times 10^{-4}$
LnL	0.700	false	false	false	false	0.700	$3.357(14) \times 10^{-4}$
LnL	0.800	true	true	false	false	0.784	$3.795(15) \times 10^{-4}$
MLP	0.912	false	false	false	false	0.912	$3.900(15) \times 10^{-4}$
LnL	0.800	true	false	false	false	0.800	$4.219(15) \times 10^{-4}$
LnL	0.800	false	false	false	false	0.800	$4.269(16) \times 10^{-4}$
MLP	0.957	false	false	false	false	0.957	$5.332(17) \times 10^{-4}$
LnL	0.900	false	false	false	false	0.900	$5.605(18) \times 10^{-4}$
MLP	0.983	false	false	false	false	0.983	$7.256(20) \times 10^{-4}$

Table L.1.: Mean background rates in $\text{keV}^{-1}\,\text{cm}^{-2}\,\text{s}^{-1}$ over the entire chip comparing all different $\ln\mathcal{L}$ and MLP setups in an energy range from 0.2 to 8 keV. Due to its efficiency at lower energies, where the majority of background is, the MLP produces the lowest mean rates.

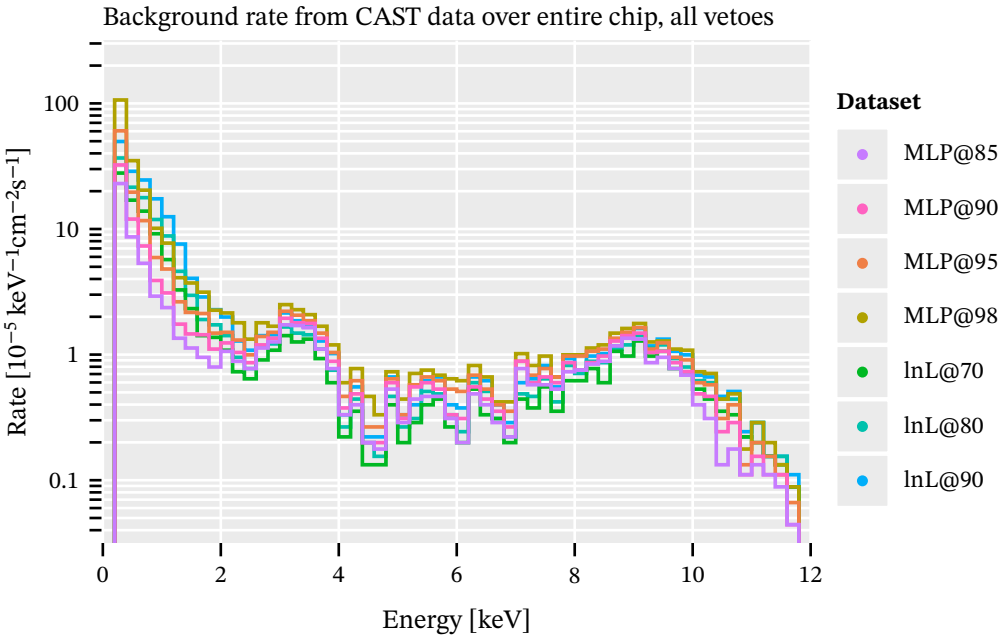


Figure L.3.: Background rate over the entire center chip of all CAST data, comparing the $\ln \mathcal{L}$ cut method and the MLP at different software efficiencies including all vetoes.

M. Background interpolation chip cutout correction Appendix

As mentioned in the main text (sec. 12.8.8), when performing the background interpolation, we need to correct for the fact that towards the edges of the chip, part of the circle (the x-y plane component) will be cut off. To correct for this we want to compute the area of the circle that is still contained on the chip. By scaling the intensity I by the missing area we correct for this. Let's discuss the actual calculation of the (possibly) double cut out circle.

The area of a single circle segment can be written as

$$A = R^2/2 \cdot (\vartheta - \sin(\vartheta))$$

where R is the radius of the circle and ϑ the angle that cuts off the circle. See the upper center part of fig. M.1 for the area A.

In the general case we need to know the area of a circle that is cut off from 2 sides, with angles ϑ_1 and ϑ_2, which are orthogonal to another. See the middle left part of fig. M.1 to see areas A and B cut off, leaving area E as the remaining area contained on the chip. For a point (x, y), define the distance to the edge of the chip to be $(\Delta x, \Delta y)$ for each axis. Then, the cutout areas A and B are given by

$$A = R^2/2 \cdot (\vartheta - \sin(\vartheta_1))$$

and

$$B = R^2/2 \cdot (\vartheta - \sin(\vartheta_2))$$

where the angles ϑ_1 and ϑ_2 (see center of fig. M.1) are related to the distances to the edge of the chip by

$$\vartheta_1 = 2 \arccos(\Delta x/R)$$
$$\vartheta_2 = 2 \arccos(\Delta y/R).$$

By subtracting areas A and B from the total area F, we remove too much however. So we need to add back:

- another circle segment D, of the angle α given by the lines connecting to the ends up each cut off line (see the center row of fig. M.1)
- the area of the triangle C, see the bottom row of fig. M.1.

The angle α relates to angles ϑ_1 and ϑ_2 via

$$\alpha = \frac{\vartheta_2}{2} - \left(\pi - \frac{\vartheta_1}{2}\right),$$

see the center row of fig. M.1.

To calculate the area C, we need the catheti of the triangle, x' and y'. See the bottom part of fig. M.1. These are the distances from the *orthogonal* cutoff line to the edge of the circle, as hopefully clear in the figure. Given that we know the center position of the circle (as that is the interpolation point), we can express x' and y' via the circle radius R and the distances from the center to the chip in each axis, Δx and Δy (they may in theory be negative if the center is outside the chip).

They are thus

$$x' = \cos\beta \cdot R - \Delta x$$
$$y' = \cos\gamma \cdot R - \Delta y,$$

where γ is the same angle as in the middle row of the schematic and β is the equivalent for ϑ_2, $\beta = \frac{\vartheta_2}{2} - \frac{\pi}{2}$.

In combination then the area E can then be expressed as:

$$E = F - A - B + C + D,$$

with F being the total area of the circle. This finally means to correct the background interpolation at a point close to the chip edges, we adjust I by

$$I'(x, y) = I \cdot \frac{F}{E(x, y)},$$

where we emphasize that $E(x, y)$ depends on the position, before normalizing with the weight W.

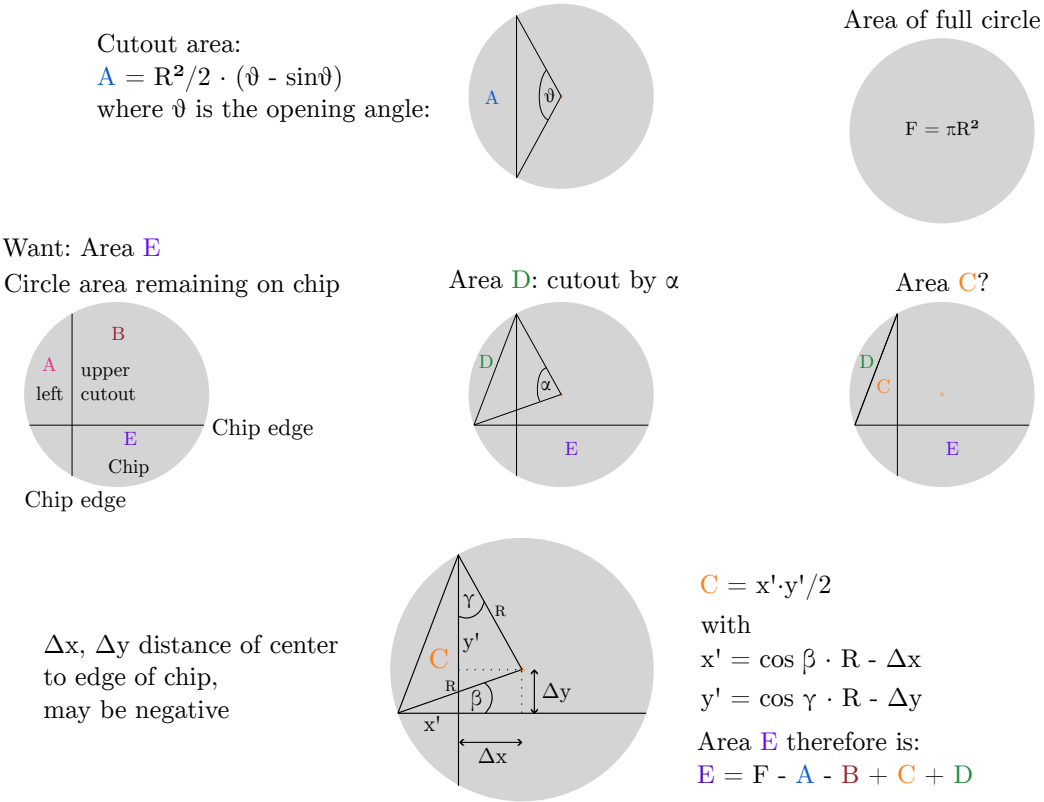


Figure M.1.: Explanation of the different areas appearing in the calculation. We want to calculate area *E*, the area remaining on the chip. Subtracting *A* and *B* makes us subtract areas *C* and *D* twice.

N. Additional limit information Appendix

N.1. Conversion probability as a function of mass Appendix

Fig. N.1 shows how the axion-photon conversion probability changes as a function of the axion mass. This implements eq. 3.2, reproduced here with slightly changed notation,

$$P_{a\mapsto\gamma}(z) = \left(\frac{g_{a\gamma}BL}{2}\right)^2 \left(\frac{\sin\left(\frac{qL}{2}\right)}{\frac{qL}{2}}\right)^2,$$

with $q = \frac{m_\gamma^2 - m_a^2}{2E_a}$. E_a is the energy of the axion (or in the context of a limit calculation the energy of a candidate). In the vacuum setup $m_\gamma = 0$. The figure shows this conversion probability for different axion energies and based on the CAST magnet. We see that the conversion probability starts falling off roughly around $m_a \approx 0.01\,\text{eV}$, with the exact value depending on energy (and personal ΔP cutoff).

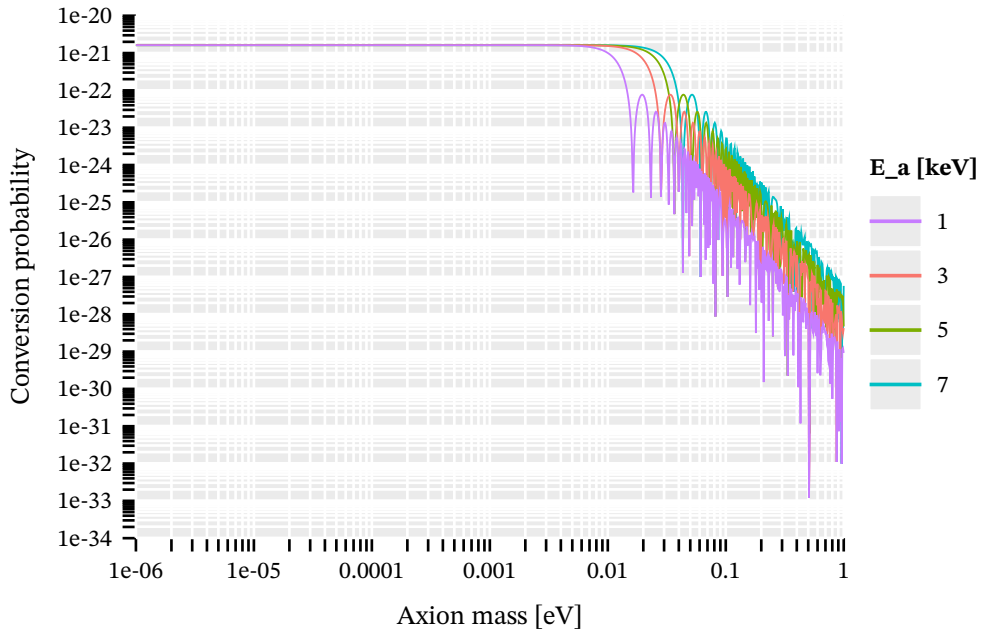


Figure N.1.: Axion-photon conversion probability as a function of axion mass. Using $g_{a\gamma} = 10^{-12}\,\text{GeV}^{-1}, B = 8.8\,\text{T}, L = 9.26\,\text{m}$. Different axion energies indicated by color.

N.2. Expected limit table with percentiles

The table shown in tab. N.1 shows the same table as tab. 12.2 in the main part of the thesis, but with a focus on the variation of the limits. The focus is on different percentiles of the distribution of sampled toy limits. The P_i columns correspond to the limit at the i^{th} percentile of all toy limits. P_{50} would be the median and thus expected limit. The table yields insight into the probabilities with which limits are expected for certain setups, given the pure statistical fluctuation possible by the measured candidates.

The veto information has been merged into the 'Type' column. A suffix (L) indicates 'line veto', 'S' the 'septem veto' and 'SL' both vetoes. '-' means no vetoes. FADC and scintillators are implicitly included if any of septem or line vetoes are in use. The units are excluded in the column names to save space. For the axion-electron and axion-photon tables they are all in GeV^{-1}. For the expected limit (last column) the uncertainty is again a bootstrapped standard deviation.

The same table for the expected axion-photon limit and chameleon limits are tab. N.2 and tab. N.3, respectively. They only have a single row, because we only computed the expected limit for one veto setup.

ε_{eff}	nmc	Type	$\varepsilon_{\text{total}}$	P_5	P_{16}	P_{25}	P_{75}	P_{84}	P_{95}	Expected
0.98	1000	MLP -	0.98	6.44e-23	6.82e-23	7.09e-23	8.65e-23	9.09e-23	1.03e-22	7.805(37)e-23
0.91	1000	MLP -	0.91	6.59e-23	6.96e-23	7.21e-23	8.75e-23	9.25e-23	1.03e-22	7.856(43)e-23
0.95	1000	MLP -	0.95	6.53e-23	6.87e-23	7.14e-23	8.74e-23	9.18e-23	1.02e-22	7.860(51)e-23
0.95	2500	MLP L	0.8	6.77e-23	7.07e-23	7.26e-23	8.72e-23	9.17e-23	1.03e-22	7.862(29)e-23
0.98	15000	MLP L	0.82	6.7e-23	7e-23	7.2e-23	8.72e-23	9.2e-23	1.02e-22	7.868(11)e-23
0.95	50000	MLP L	0.8	6.75e-23	7.04e-23	7.25e-23	8.72e-23	9.18e-23	1.02e-22	7.8782(65)e-23
0.95	15000	MLP L	0.8	6.75e-23	7.04e-23	7.24e-23	8.72e-23	9.16e-23	1.03e-22	7.879(12)e-23
0.98	2500	MLP L	0.82	6.73e-23	7.01e-23	7.19e-23	8.72e-23	9.22e-23	1.02e-22	7.883(30)e-23
0.86	1000	MLP -	0.86	6.74e-23	7.08e-23	7.31e-23	8.88e-23	9.35e-23	1.03e-22	7.960(51)e-23
0.91	2500	MLP L	0.76	6.91e-23	7.18e-23	7.38e-23	8.9e-23	9.3e-23	1.03e-22	7.99(16)e-23
0.91	15000	MLP L	0.76	6.9e-23	7.18e-23	7.38e-23	8.87e-23	9.34e-23	1.04e-22	8.004(11)e-23
0.98	2500	MLP SL	0.76	6.93e-23	7.2e-23	7.42e-23	8.97e-23	9.47e-23	1.06e-22	8.085(29)e-23
0.95	2500	MLP S	0.78	6.91e-23	7.22e-23	7.43e-23	9.08e-23	9.53e-23	1.07e-22	8.113(36)e-23
0.95	2500	MLP SL	0.73	6.99e-23	7.29e-23	7.49e-23	9e-23	9.46e-23	1.05e-22	8.125(31)e-23
0.98	2500	MLP S	0.8	6.82e-23	7.16e-23	7.42e-23	9.02e-23	9.46e-23	1.06e-22	8.131(32)e-23
0.86	2500	MLP L	0.72	7.03e-23	7.32e-23	7.54e-23	9.09e-23	9.58e-23	1.06e-22	8.156(30)e-23
0.86	15000	MLP L	0.72	7.03e-23	7.32e-23	7.54e-23	9.06e-23	9.51e-23	1.06e-22	8.183(13)e-23
0.91	2500	MLP S	0.74	7.03e-23	7.33e-23	7.54e-23	9.12e-23	9.63e-23	1.07e-22	8.22(19)e-23
0.9	2500	LnL L	0.75	6.96e-23	7.28e-23	7.49e-23	9.13e-23	9.61e-23	1.06e-22	8.217(37)e-23
0.91	2500	MLP SL	0.7	7.1e-23	7.42e-23	7.62e-23	9.17e-23	9.64e-23	1.08e-22	8.287(33)e-23
0.86	2500	MLP S	0.7	7.19e-23	7.5e-23	7.72e-23	9.27e-23	9.71e-23	1.08e-22	8.401(29)e-23

Continued on next page

Continued from previous page

ε_{eff}	nmc	Type	$\varepsilon_{\text{total}}$	P_5	P_{16}	P_{25}	P_{75}	P_{84}	P_{95}	Expected
0.9	2500	LnL SL	0.69	7.21e-23	7.52e-23	7.74e-23	9.38e-23	9.89e-23	1.11e-22	8.427(34)e-23
0.86	2500	MLP SL	0.66	7.32e-23	7.6e-23	7.79e-23	9.38e-23	9.76e-23	1.08e-22	8.459(35)e-23
0.8	2500	LnL L	0.67	7.3e-23	7.6e-23	7.83e-23	9.4e-23	9.91e-23	1.09e-22	8.499(32)e-23
0.9	2500	LnL -	0.9	6.91e-23	7.43e-23	7.73e-23	9.57e-23	1.01e-22	1.12e-22	8.579(37)e-23
0.8	2500	LnL -	0.8	7.13e-23	7.59e-23	7.88e-23	9.79e-23	1.03e-22	1.15e-22	8.738(39)e-23
0.8	2500	LnL SL	0.62	7.52e-23	7.82e-23	8.03e-23	9.68e-23	1.02e-22	1.13e-22	8.747(41)e-23
0.7	2500	LnL L	0.59	7.72e-23	8.02e-23	8.21e-23	9.86e-23	1.04e-22	1.16e-22	8.930(40)e-23
0.7	2500	LnL -	0.7	7.4e-23	7.87e-23	8.23e-23	1.01e-22	1.07e-22	1.19e-22	9.086(33)e-23
0.7	2500	LnL SL	0.54	8.01e-23	8.28e-23	8.51e-23	1.02e-22	1.08e-22	1.2e-22	9.257(35)e-23

Table N.1.: Table of the expected limits for different veto setups, comparable to tab. 12.2, with a focus on the percentiles P_i of the computed toy limits. For example P_{25} is the 25$^{\text{th}}$ percentile of the distribution of toy limits. All values in units of GeV^{-1}.

ε_{eff}	nmc	Type	$\varepsilon_{\text{total}}$	P_5	P_{16}	P_{25}	P_{75}	P_{84}	P_{95}	Expected
0.95	10000	MLP L	0.8	8.24e-11	8.5e-11	8.66e-11	9.56e-11	9.83e-11	1.04e-10	9.0650(75)e-11

Table N.2.: Table of the different percentiles for the single axion-photon expected limit. All values in units of GeV^{-1}.

ε_{eff}	nmc	Type	$\varepsilon_{\text{total}}$	P_5	P_{16}	P_{25}	P_{75}	P_{84}	P_{95}	Expected
0.95	10000	MLP L	0.8	3.22e+10	3.35e+10	3.43e+10	3.82e+10	3.93e+10	4.16e+10	3.6060(39)e+10

Table N.3.: Table of the different percentiles for the single chameleon expected limit.

N.3. Observed limit - axion photon $g_{a\gamma}$

Fig. N.2 shows the sampled coupling constants in $g_{a\gamma}^4$ of the calculation for the observed limit, i.e. the marginal posterior likelihood function for the real candidates for the axion-photon coupling.

N.4. Observed limit - chameleon β_γ

Fig. N.3 shows the sampled coupling constants in β_γ^4 of the calculation for the observed limit, i.e. the marginal posterior likelihood function for the real candidates for the chameleon coupling.

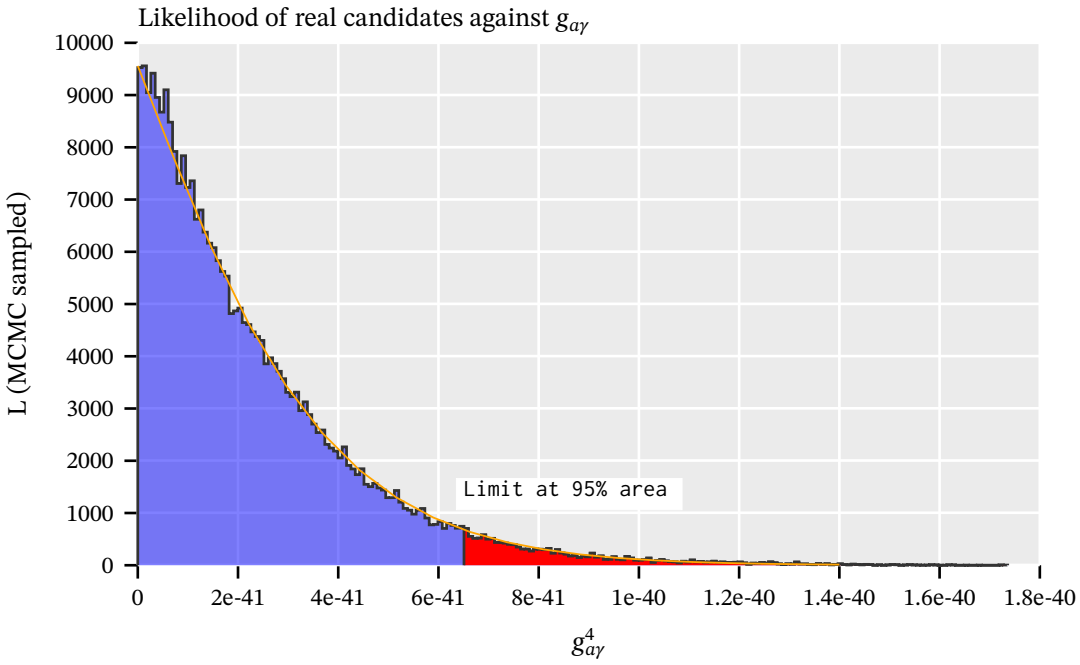


Figure N.2.: Marginal posterior likelihood function of the real candidates for the axion-photon coupling constant in $g^4_{a\gamma}$ space. The yellow line is a numerical integration of the likelihood function using Romberg's method [182]. Limit at $g^4 \approx 6.56 \times 10^{-41}\,\text{GeV}^{-4} \Rightarrow g \approx 9 \times 10^{-11}\,\text{GeV}^{-1}$.

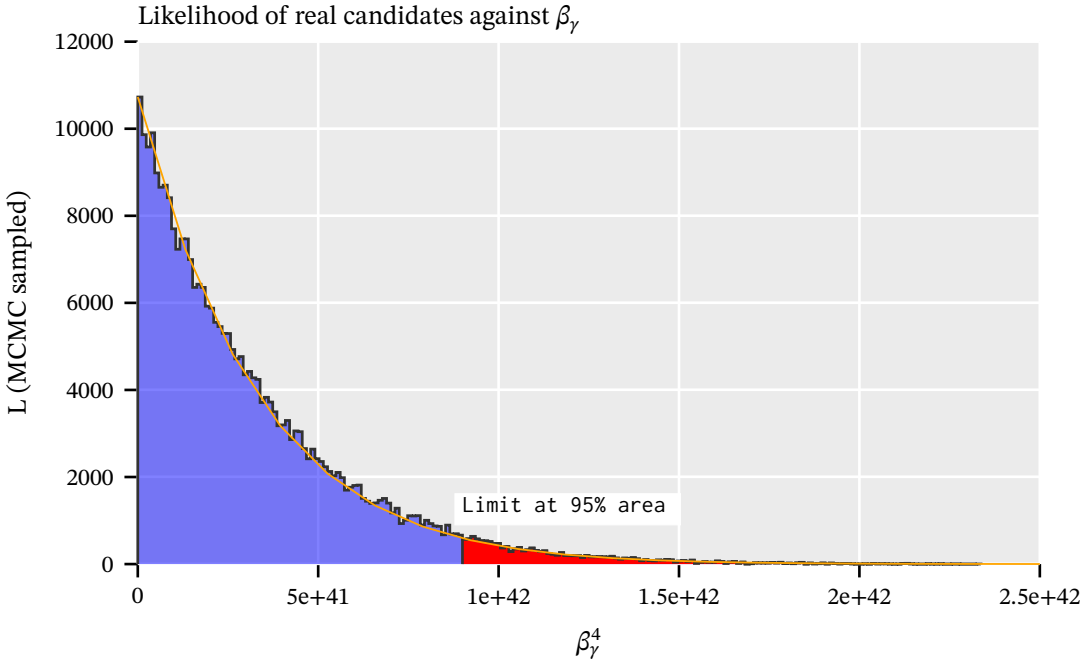


Figure N.3.: Marginal posterior likelihood function of the real candidates for the chameleon coupling constant in β^4_γ space. The yellow line is a numerical integration of the likelihood function using Romberg's method [182]. Limit at $\beta^4 \approx 9.2 \times 10^{41} \Rightarrow \beta \approx 3.1 \times 10^{10}$.

O. Software Software:Appendix

Contents

In this appendix we will go over the software developed and used in the course of this thesis with a bit more detail than in the main thesis (beginning of chapter 8). The focus is more on the technical and usability side and not the physics application. Read this appendix either if you

- are simply interested,
- are intending to reproduce the analysis or use (parts of) the software for your own data,
- wish to further process data produced by these tools with your own software.

O.1. TimepixAnalysis

Introduced in the main part, in sec. 8.1, `TimepixAnalysis` [194] is the name for the repository containing a large collection of different programs for the data reconstruction and analysis of Timepix based detectors.

Generally, the `README` in the repository gives an overview of all the relevant programs, installation instructions and more. For further details therefore check there or simply open an issue in the repository [194].

Here we will now go over the main programs required to handle the Septemboard data taken at CAST so that in appendix P we can present the commands to for the entire CAST data reconstruction.

> Note: in the PDF and HTML version of the thesis I provide some links to different parts of the repository. These generally point to Github, because the main public repository of `TimepixAnalysis` is found there. This is mostly out of convenience though. It should be straightforward to map them to the paths inside of your local copy of the repository, but that makes it trickier to link to.

O.1.1. Common points between all TimepixAnalysis programs

All programs in the TimepixAnalysis repository are command line only. While it would be quite doable to merge the different programs into a single graphical user interface (GUI), I'm personally not much of a GUI person. Each program usually has a large number of (optional) parameters. Keeping the GUI up to date with (in the past, quickly) changing features is just extra work, which I personally did not have any use for (if someone wishes to write a GUI for TimepixAnalysis, I'd be more than happy to mentor though).

Every program uses `cligen` [1], a command line interface generator. Based on the definition of the main procedure(s) in the program, a command line interface is generated. While `cligen` provides extremely simplified command line argument parsing for the developer, it also gives a nice `help` screen for every program. For example, running the first program of the analysis pipeline `raw_data_manipulation` with the `-h` or `--help` option:

```
raw_data_manipulation -h
```

yields the help screen as shown in listing 16 [2]. Keep this in mind, if you are unsure about how to use any of the here mentioned programs.

[1] `https://github.com/c-blake/cligen`

[2] You can have a `$HOME/.config/cligen/config` configuration file to adjust the output style (color, column widths, drop entire columns etc.).

Further, there is a TOML configuration file in the repository (`Analysis/ingrid/config.toml` from the repository root), which controls many aspects of the different programs. Most of these can be overwritten by command line arguments to the appropriate programs and some also via environment variables. See the extended thesis for information about this, it is mentioned where important.

O.1.2. Dependencies

TimepixAnalysis mainly has a single noteworthy external dependency, namely the HDF5 [214] library. The vast majority of code (inside the repository itself and its dependencies) is pure Nim. Two optimization libraries written in C `mpfit` (Levenberg-Marquardt) [3] and NLopt [4] are wrapped from Nim and are further minor dependencies. Local compilation and installation of these is trivial and explained in the TimepixAnalysis README.

For those programs related to the multilayer perceptron (MLP) training or usage, PyTorch [162] is an additional dependency via [198]. Flambeau installs a suitable PyTorch version for you.

Common other dependencies are the `cairo` graphics library [5] and a working BLAS and LAPACK installation.

O.1.3. Compilation

Nim being a compiled language means we need to compile the programs mentioned below. It can target a C or C++ backend (among others). The compilation commands differ slightly between the different programs and can depend on usage. The `likelihood` program below for example can be compiled for the C or C++ backend. In the latter case, the MLP as a classifier is compiled in.

Generally, compilation is done via:

```
nim c -d:release foo.nim
```

where `foo.nim` is any of the programs below. `-d:release` tells the Nim compiler to compile with optimizations (you can compile with `-d:danger` for even faster, but less safe, code). Replace `c` by `cpp` to compile to the C++ backend.

See the TimepixAnalysis README for further details on how to compile each program.

Unless otherwise specified, each program mentioned below is located in `Analysis/ingrid` from the root of the TimepixAnalysis repository.

[3] `https://pages.physics.wisc.edu/~craigm/idl/cmpfit.html`
[4] `https://nlopt.readthedocs.io/en/latest/`
[5] `https://cairographics.org/download/`

```
Usage:
  main [REQUIRED,optional-params]
Version: 44c0c91 built on: 2023-12-06 at 13:01:35
Options:
  -h, --help                              print this cligen-erated help

  --help-syntax                           advanced: prepend,plurals,..

  -p=, --path=        string      REQUIRED set path

  -r=, --runType=     RunTypeKind REQUIRED Select run type (Calib | Back | Xray)
                                           The following are parsed case
                                           insensetive:
                                             Calib = {"calib", "calibration", "c"}
                                             Back = {"back", "background", "b"}
                                             Xray = {"xray", "xrayfinger", "x"}

  -o=, --out=         string      ""       Filename of output file. If none
                                           given will be set to run_file.h5.

  -n, --nofadc        bool        false    Do not read FADC files.

  -i, --ignoreRunList bool        false    If set ignores the run list 2014/15
                                           to indicate using any rfOldTos run

  -c=, --config=      string      ""       Path to the configuration file to use.
                                           Default is config.toml in directory of
                                           this source file.
  ...

  -t, --tpx3          bool        false    Convert data from a Timepix3 H5 file
                                           to TPA format instead of a Tpx1 run
                                           directory
  ...
```

Listing 16: Example help output of the `raw_data_manipulation` program when run with `-h` or `--help` (some options were removed here to shorten the output).

O.1.4. raw_data_manipulation

raw_data_manipulation is the first step of the analysis pipeline. Essentially, it is a parsing stage of the data generated by TOS (see section A.2.1 for an explanation of it) and storing it in a compressed HDF5 [214] data file.

The program is fed a directory containing a TOS run via the -p / --path argument. Either a directory containing a single run (i.e. a data taking period ranging typically from minutes to days in length), or a directory that itself contains multiple TOS run directories. Runs compressed as gzipped TAR balls, .tar.gz are also supported.

All data files contained in a run directory will then be parsed in a multithreaded way. The files are memory mapped and parsed in parallel into a Run data structure, which itself contains Event structures.

If FADC files are present in a directory, these will also be parsed into FadcEvent structures in a similar fashion, unless explicitly disabled via the --nofadc option.

Each run is then written into the output HDF5 file as a 'group' (HDF5 terminology). The meta data about each run and event are stored as 'attributes' and additional 'datasets', respectively. The structure of the produced HDF5 file is shown in sec. O.1.4.

In addition, the tool also supports input from HDF5 files containing the raw data from a Timepix3 detector. That data is parsed and reprocessed into the same kind of file structure.

HDF5 data layout generated by raw_data_manipulation

The listing 17 shows the layout of the data stored in the HDF5 files after the raw_data_manipulation program has processed the TOS run folders. The data is structured in groups based on each run, chip and the FADC (if available). Generally each "property" is stored in its own dataset for performance reasons to allow faster access to individual subsets of the data (read only the hits, only x/y data, etc.). While HDF5 supports even heterogeneous compound datasets (that is different data types in different "columns" of a 2D like dataset), these are only used sparingly and not at all in the raw_data_manipulation output, as reading individual columns from these is inefficient.

O.1.5. reconstruction

After the raw data has been converted to storage in HDF5, the reconstruction tool is used to start the actual analysis of the data. The program receives an input HDF5 file via the -i / --input argument. As the name implies, the first stage of data analysis is in the form of reconstructing the basic properties of each event. In this stage all events are processed in a multithreaded way. The steps for cluster finding and geometric cluster reconstruction (as mentioned in sec. 8.4) are performed and the data is written to the desired output file given by -o / --outfile.

```
├ runs
  ├ run_<number>
    ├ chip_0 # one for each chip in the event
      ├ Hits # number of hits in each event
      ├ Occupancy # a 2D occupancy map of this run
      ├ ToT # all ToT values of this run
      ├ raw_ch # the ToT/ToA values recorded for each event (ragged data)
      ├ raw_x # the x coordinates recorded for each event (ragged data)
      ├ raw_y # the y coordinates recorded for each event (ragged data)
    ├ chip_i # all other chips
      ├ ...
    ├ fadc # if available
      ├ eventNumber # event number of each entry
                    # (not all events have FADC data)
      ├ raw_fadc # raw FADC data (uncorrected, all 10240 registers)
      ├ trigger_record # temporal correction factor for each event
    ├ fadcReadout # flag if FADC was readout in each event
    ├ fadcTriggerClock # clock cycle FADC triggered
    ├ szintillator trigger clocks # datasets for each scintillator
    ├ timestamp # timestamp of each event
  ├ run_i # all other runs
    ├ ...
```

Listing 17: An abstract overview of the general layout of the generated HDF5 files. Each entry shown that has any children is an HDF5 group. Every leaf node is an HDF5 dataset. The data is ordered by availability. Each run is a separate group. Within all chips have their own groups with data associated to that chip. The common datasets are those that contain data from the TOS event header. FADC data stored as the raw memory dumps from the FADC files (for which fewer than regular TOS data files exist).

The produced output HDF5 file then also acts as the *input* file for `reconstruction` for all further, optional reconstruction steps. These are mentioned at different parts in the thesis, but we will explain them shortly here now.

`--only_fadc` Performs the reconstruction of the FADC data to calculate FADC values such as rise and fall times.

`--only_fe_spec` If the input file contains ^{55}Fe calibration runs, creates the ^{55}Fe spectra and performs fits to them. Also performs the energy calibration for each run.

`--only_charge` Performs the `ToT` calibration of all runs to compute the detected charges in electrons. Requires each chip to be present in the InGrid database (see sec. O.1.10).

`--only_gas_gain` Computes the gas gain in the desired interval lengths via Pólya fits.

`--only_gain_fit` If the input file contains ^{55}Fe calibration runs, performs the fit of energy calibration runs against the gas gain of each interval. Required to perform energy calibration in background runs.

`--only_energy_from_e` Performs the energy calibration for each cluster in the input file.

HDF5 data layout generated by `reconstruction`

The HDF5 file generated by `reconstruction` follows closely the one from `raw_data_manipulation`. The main difference is that within each chip group now each chip has a different number of entries in the datasets, as each entry now corresponds to a single cluster, not an event from the detector. On some events multiple clusters on a single chip may be reconstructed, while other events may be fully empty. This means an additional `eventNumber` dataset is required for each chip, which maps back each cluster to a corresponding event.

Aside from that the other major difference is simply that each chip has a larger number of datasets, as each computed cluster property is a single variable. Also additional new datasets will be created during the data calibration (charge calibration, computation of the gas gain, etc.).

Listing 18 shows the layout in a similar fashion to the equivalent for `raw_data_manipulation` before.

O.1.6. `cdl_spectrum_creation`

This is a helper program responsible for the treatment of the X-ray reference data taken at the CAST Detector Lab in Feb. 2019. It receives an HDF5 file as input that is fully reconstructed using `raw_data_manipulation`

```
├ reconstruction
  ├ run_<number>
    ├ chip_0 # one for each chip in the event
      ├ datasets for each property
      ├ optional datasets for calibrations
    ├ chip_i # all other chips
      ├ ...
    ├ fadc # if available
      ├ datasets for each FADC property
    ├ common datasets # copied from `raw_data_manipulation` input
  ├ run_i # all other runs
    ├ ...
```

Listing 18: Abstract overview of the data layout of the `reconstruction` HDF5 output. It is essentially the same layout as the `raw_data_manipulation` HDF5 files, but contains more datasets, due to larger number of properties.

and `reconstruction` containing all runs taken in the CDL. An additional Org table is used as reference to map each run to the correct target/filter kind, found in `resources/cdl_runs_2019.org`.

The program performs the fits to the correct fluorescence lines for each run based on the target/filter kind in use. It can also produce a helper HDF5 file called `calibration-cdl-2018.h5` via the `genCdlFile` argument, which contains all CDL data split by target/filter kind. This file is used in the context of the likelihood cut method to produce the reference distributions for each cluster property used.

O.1.7. `likelihood`

The `likelihood` program is the (historically named) tool that applies the classifier and any selection of vetoes to an input file. The input files are a fully reconstructed background HDF5 file, corresponding calibration runs and the `calibration-cdl-2018.h5` file mentioned above. It has a large number of command line options to adjust the classifier that is used, software efficiency, vetoes, region of the chip to cut to, whether tracking data or background data is selected and more. The program writes the remaining clusters (with additional meta information) to the HDF5 file given by the `--h5out` argument. The structure is essentially identical to that of the `reconstruction` tool (the data is stored in a `likelihood` group instead of a `reconstruction` group).

The selection of tracking or non-tracking data requires information about when solar trackings took place as attributes inside of the background HDF5 files. These are added using the `cast_log_reader`, see sec. O.1.11.

It is also used directly to estimate the random coincidences of the septem and line vetoes, as mentioned in sec. 11.5.5.

O.1.8. `determineDiffusion` directory

The `Analysis/ingrid/determineDiffusion` directory contains the library / binary to empirically determine the gas diffusion parameters from input data, as explained in sec. 11.4.3. It can either be compiled as a standalone program or be used as a library.

O.1.9. `nn` directory

The `Analysis/ingrid/nn` subdirectory in the TimepixAnalysis repository contains the programs related to the training and evaluation of the multilayer perceptrons (MLP) used in the thesis. Of note is the `train_ingrid` program, which is the program to train a network. It allows to customize the network to be trained based on command line arguments describing the number of neurons, hidden layers, optimizers, activation functions and so forth. The extended thesis contains the command to train the best performing network.

Secondly, the `simulate_xrays` program is a helper program to produce an HDF5 file containing simulated X-rays as described in sec. 11.4.2. It makes use of the `fake_event_generator.nim` file in the `ingrid` directory, which contains the actual logic.

O.1.10. `InGridDatabase`

The InGrid database is both a library part of the TimepixAnalysis repository (`InGridDatabase` directory) as well as a binary tool and the name for a very simple 'database' storing information about different GridPix chips.

At its core the 'database' part is an HDF5 file containing chip calibrations (ToT, SCurve, ...) mapped to timestamps or run numbers in which these are applicable. This allows (mainly) the `reconstruction` program to retrieve the required calibrations automatically without user input based on the given input files.

To utilize it, the `databaseTool` needs to be compiled as a binary. Chips are added to the database using this tool. A directory describing the applicable run period and containing calibration files for the chip need to follow the format seen for example in:

`https://github.com/Vindaar/TimepixAnalysis/tree/master/resources/ChipCalibrations/Run2`

for the Run-2 period of the Septemboard. The `runPeriod.toml` file describes the applicability of the data, see listing 19 for the file in this case. For each chip, there is simply a directory with the

calibration files as produced by TOS and an additional `chipInfo.txt` file, see listing 20. Note that the `runPeriod` name needs to match the name of one of the run periods listed in the TOML file.

The `databaseTool` also allows to perform fits to the calibration data, if needed (for example to analyze SCurves or the raw ToT calibration data).

```
title = "Run period 2 of CAST, 2017/18"
# list of the run periods defined in the file
runPeriods = ["Run2"]

[Run2]
start = 2017-10-30
stop = 2018-04-11
# either as a sequence of run numbers
validRuns = [ 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89
  90, 91, 92, 93, 94, 95, 96, 97, 98, 99,100,101,102,103
  104,105,106,107,108,109,110,111,112,113,114,115,116,117
  118,119,120,121,122,123,124,125,126,127,128,145,146,147
  148,149,150,151,152,153,154,155,156,157,158,159,160,161
  162,163,164,165,166,167,168,169,170,171,172,173,174,175
  176,177,178,179,180,181,182,183,184,185,186,187,188,189
]
# or as simply a range given as start and stop values
firstRun = 76
lastRun = 189
```

Listing 19: Example of a `runPeriod.toml` file.

```
chipName: H10 W69
runPeriod: Run2
board: SeptemH
chipNumber: 3
Info: This calibration data is valid for Run 2 starting until March 2018!
```

Listing 20: An example of a `chipInfo.txt` file describing a single chip for a run period.

O.1.11. `cast_log_reader`

The `LogReader/cast_log_reader` is a utility to work with the slow control and tracking log files produced by CAST. It can parse and analyze the different log file formats used over the years and provide different information (magnet operation statistics for example). For the Septemboard detector it pro-

vides the option to parse the tracking log files and add the tracking information to the background HDF5 files.

The program first parses the log files and determines valid solar trackings. Then, given an HDF5 data file containing the background data each solar tracking is mapped to a background run. One background run may have zero or more solar trackings attached to it. In the final step the solar tracking start and stop information is added to each run as additional meta data. During the later stages of processing in other `TimepixAnalysis` programs, notably `likelihood`, this meta data is then used to only consider background (non tracking) or solar tracking data.

The CAST log files relevant for the Septemboard detector can be found together with the Septemboard CAST data.

O.1.12. `mcmc_limit_calculation`

`mcmc_limit_calculation` is the 'final' tool relevant for this thesis. As the name implies it performs the limit calculation using Markov Chain Monte Carlo (MCMC) as explained in detail in chapter 12. See the extended thesis on how to use it.

O.1.13. `Tools` directory

Multitude of tools for various things that were analyzed over the years. Includes things like computing the gas properties of the Septemboard gas mixture and detection efficiency.

O.1.14. `resources` directory

A large number of resources required or simply useful about different data takings, efficiencies, log files and more, which are small enough to be part of a non LFS git repository.

O.1.15. `Plotting` directory

From a practical analysis point of view, the `Plotting` directory is one of the most interesting parts of the repository. It contains different tools to visualize data at various stages of the analysis pipeline. The most relevant are mentioned briefly here.

`plotBackgroundClusters`

`plotBackgroundCluster` produces plots of the distribution of cluster center left after application of the `likelihood` program. This is used to produce figures like 6.12 and fig. 11.29.

`plotBackgroundRate`

`plotBackgroundRate` is the main tool to visualize background (or raw data) spectra. All such plots in the thesis are produced with it. Input files are reconstructed HDF5 files or the result of the `likelihood` program.

`plotCalibration`

`plotCalibration` is a tool to produce visualizations of the different Timepix calibration steps, e.g. ToT calibration, SCurve scans and so on. The figures in sec. C.1 are produced with it.

`plotData`

`plotData` is a very versatile tool to produce a variety of different plots. It can produce histograms of the different geometric properties, occupancy maps, event displays and more. If desired, it can produce a large number of plots for an input data file in one go. It is very powerful, because it can receive an arbitrary number of cuts on any dataset present in the input. This allows to produce visualizations for any desired subset of the data. For example to produce event displays or histograms for only those events with specific geometric properties. It is an exceptionally useful tool to understand certain subsets of data that appear 'interesting'.

In sec. K.2 we mention non noisy FADC events in a region of the rise time / skewness space of fig. K.7. These are easily filtered to and investigated using `plotData`. Also fig. K.10 and fig. 8.3 are produced with it, among others. Generally, it favors information (density) over aesthetically pleasing visualizations though.

O.2. Other libraries relevant for TimepixAnalysis

A few other libraries not part of the TimepixAnalysis repository bear mentioning, due to their importance. They were written alongside TimepixAnalysis.

ggplotnim A `ggplot2` [6] inspired plotting library. All plots in this thesis are produced with it.

Datamancer A `dpylr` [7] inspired data frame library.

nimhdf5 A high level interface to the HDF5 [214] library, somewhat similar to `h5py` [8] for Python.

[6] `https://ggplot2.tidyverse.org/`
[7] `https://dplyr.tidyverse.org/`
[8] `https://www.h5py.org/`

Unchained A library to perform zero runtime overhead, compile-time checking and conversion of physical units. Exceptionally useful to avoid bugs due to wrong unit conversions and a big help when dealing with natural units.

Measuremancer A library to deal with measurements with uncertainties. Performs automatic Gaussian error propagation when performing calculations with measurements.

xrayAttenuation A library dealing with the interaction of X-rays with matter (gases and solids). It is used to calculate things like absorption of gas and reflectivities of the X-ray telescope in this thesis.

TrAXer The raytracer used to compute the axion image, expanded on in appendix R.

P. Full data reconstruction Appendix

We will now go over how to fully reconstruct all CAST data from the Run-2 and Run-3 data taking campaigns all the way up to the computation of the observed limit. This assumes the following:

- all CAST data files are located in a directory accessible by a `DATA` environment variable. Inside should be the directory structure containing directories `2017`, `2018`, `2018_2` and `CDL_2019` as downloaded from Zenodo [191].
- all binaries used below are compiled and their location is in your `PATH` variable. There is a `bin` directory in the repository root, which contains symbolic links to the binaries if they are placed next to the source file. I recommend adding this directory to your `PATH`.

  Note: There is also the `Analysis/ingrid/runAnalysisChain` program, which automates a majority of these calls. See the section below on how it replaces the data parsing and reconstruction. Here I wish to illustrate the actual steps and don't hide any.

Furthermore, the commands below produce the correct results assuming the `Analysis/ingrid/config.toml` file is used as committed as part of the `phd` tag in the TimepixAnalysis repository.

P.1. Raw data parsing and reconstruction

If this is all in place, first perform the raw data parsing.

Run-2, background and calibration:

```
raw_data_manipulation -p $DATA/2017/DataRuns \
                      --runType rtBackground \
                      --out $DATA/DataRuns2017_Raw.h5
raw_data_manipulation -p $DATA/2017/CalibrationRuns \
                      --runType rtCalibration \
                      --out $DATA/DataRuns2017_Raw.h5
```

Run-3:

```
raw_data_manipulation -p $DATA/2018_2/DataRuns \
                      --runType rtBackground \
                      --out $DATA/DataRuns2018_Raw.h5
raw_data_manipulation -p $DATA/2018_2/CalibrationRuns \
                      --runType rtCalibration \
                      --out $DATA/DataRuns2018_Raw.h5
```

Next, the initial data reconstruction (geometric properties):

Run-2:

```
reconstruction -i $DATA/DataRuns2017_Raw.h5 \
               -o $DATA/DataRuns2017_Reco.h5
reconstruction -i $DATA/CalibrationRuns2017_Raw.h5 \
               -o $DATA/CalibrationRuns2017_Reco.h5
```

Run-3:

```
reconstruction -i $DATA/DataRuns2018_Raw.h5 \
               -o $DATA/DataRuns2018_Reco.h5
reconstruction -i $DATA/CalibrationRuns2018_Raw.h5 \
               -o $DATA/CalibrationRuns2018_Reco.h5
```

Now, the next steps of the reconstruction (charge calibration, gas gain and FADC reconstruction):

```
DATA=~/CastData/data
for typ in Data Calibration; do
    for year in 2017 2018; do
        file="${DATA}/${typ}Runs${year}_Reco.h5"
        reconstruction -i $file --only_charge
        reconstruction -i $file --only_fadc
        reconstruction -i $file --only_gas_gain
    done
done
```

where we simply loop over the `Data` and `Calibration` prefixes and years.

With this done, we can perform the ^{55}Fe calibration fits:

```
reconstruction -i $DATA/CalibrationRuns2017_Reco.h5 --only_fe_spec
reconstruction -i $DATA/CalibrationRuns2018_Reco.h5 --only_fe_spec
```

and then finally the fit of energy calibration factors determined from each fit against its gas gain:

```
reconstruction -i $DATA/CalibrationRuns2017_Reco.h5 --only_gain_fit
reconstruction -i $DATA/CalibrationRuns2018_Reco.h5 --only_gain_fit
```

This then allows to calibrate the energy for each file:

```
reconstruction -i $DATA/DataRuns2017_Reco.h5 --only_energy_from_e
reconstruction -i $DATA/DataRuns2018_Reco.h5 --only_energy_from_e
reconstruction -i $DATA/CalibrationRuns2017_Reco.h5 --only_energy_from_e
reconstruction -i $DATA/CalibrationRuns2018_Reco.h5 --only_energy_from_e
```

P.2. Parse and reconstruct the CDL data

In order to use the likelihood cut method we also need to parse and reconstruct the CDL data. Note that this can be done before even reconstructing any of the CAST data files (with the exception of `--only_energy_from_e`, which is optional anyway).

```
raw_data_manipulation -p $DATA/CDL_2019/ -r Xray -o $DATA/CDL_2019/CDL_2019_Raw.h5
reconstruction -i $DATA/CDL_2019/CDL_2019_Raw.h5 -o $DATA/CDL_2019/CDL_2019_Reco.h5
reconstruction -i $DATA/CDL_2019/CDL_2019_Reco.h5 --only_charge
reconstruction -i $DATA/CDL_2019/CDL_2019_Reco.h5 --only_fadc
reconstruction -i $DATA/CDL_2019/CDL_2019_Reco.h5 --only_gas_gain
reconstruction -i $DATA/CDL_2019/CDL_2019_Reco.h5 --only_energy_from_e
```

With this file we can then run `cdl_spectrum_calibration` in order to produce the `calibration-cdl-2018.h5` file:

```
cdl_spectrum_creation $DATA/CDL_2019/CDL_2019_reco.h5 \
                      --genCdlFile --year=2018
```

P.3. Add tracking information to background files

To add the tracking information, the CAST log files are needed. The default path (which may be a symbolic link of course) is in `resources/LogFiles/tracking-logs` from the TimepixAnalysis root.

```
for year in 2017 2018; do
    cast_log_reader tracking \
                -p $TPXDIR/resources/LogFiles/tracking-logs \
                --startTime "2017/01/01" \
                --endTime "2018/12/31" \
```

```
6                 --h5out "${DATA}/DataRuns${year}_Reco.h5"
7 done
```

Here we assume `TPXDIR` is an environment variable to the root of the TimepixAnalysis repository. Feel free to adjust if your path differs.

P.4. Using `runAnalysisChain`

All of the commands above can also be performed in one by using `runAnalysisChain`:

```
./runAnalysisChain \
    -i $DATA \
    --outpath $DATA \
    --years 2017 --years 2018 \
    --calib --back --cdl \
    --raw --reco \
    --logL \
    --tracking
```

which tells the `runAnalysisChain` helper to parse all data files for Run-2 and Run-3, do the same for the CDL data files, reconstruct with all steps, compute the likelihood values and add the tracking information to the background data files. It makes the same assumptions about the location of the data files as in the above.

In particular, if the tracking log files are in a different location, add the `--trackingLogs` argument to the above with the correct path.

P.5. Applying a classifier

To apply a classifier we use the `likelihood` program. We will show how to apply it for one file now here. Generally, manual calling of `likelihood` is not needed, more on that below.

Let's apply the likelihood cut method using all vetoes for the Run-2 data:

```
likelihood \
    -f $DATA/DataRuns2017_Reco.h5 \
    --h5out $DATA/classifier/run2_lnL_80_all_vetoes.h5 \
    --region=crAll \
    --cdlYear=2018 \
    --lnL --signalEfficiency=0.8 \
    --scintiveto --fadcveto --septemveto --lineveto \
    --vetoPercentile=0.99 \
```

```
    --cdlFile=$DATA/CDL_2019/calibration-cdl-2018.h5 \
    --calibFile=$DATA/CalibrationRuns2017_Reco.h5
```

which should be mostly self explanatory. The `--signalEfficiency` argument is the software efficiency of the likelihood cut. `--vetoPercentile` refers to the FADC veto cut. The method is applied to the entire center chip (`--region=crAll`). A `--tracking` flag could be added to apply the classifier only to the tracking data (in this case it is applied only to *non* tracking data).

As explained in sec. 11.6, many different veto setups were considered. For this reason a helper tool `createAllLikelihoodCombinations` (excuse the verbose name, heh) exists to simplify the process of calling `likelihood`. It is located in the `Analysis` directory.

For example:

```
./createAllLikelihoodCombinations \
    --f2017 $DATA/DataRuns2017_Reco.h5 \
    --f2018 $DATA/DataRuns2018_Reco.h5 \
    --c2017 $DATA/CalibrationRuns2017_Reco.h5 \
    --c2018 $DATA/CalibrationRuns2018_Reco.h5 \
    --regions crAll \
    --vetoSets "{fkMLP, fkFadc, fkScinti, fkSeptem, fkLineVeto, fkExclusiveLineVeto}" \
    --mlpPath $MLP/<mlp_of_choice>.pt \
    --fadcVetoPercentile 0.99 \
    --signalEfficiency 0.85 --signalEfficiency 0.90 \
    --signalEfficiency 0.95 --signalEfficiency 0.98 \
    --out $DATA/classifier/mlp \
    --cdlFile $DATA/CDL_2019/calibration-cdl-2018.h5 \
    --multiprocessing \
    --jobs 6 \
    --dryRun
```

would apply the MLP classifier (here with a dummy name) with successive additions of all vetoes in `vetoSets` at 4 different signal efficiencies and store the HDF5 output files in `DATA/classifier/mlp`. Multiple `vetoSets` arguments can be given in one call. Also each `fk` entry can be prefixed with a `+` to indicate that this veto should not be run on its own. This allows flexibility. As indicated by `--multiprocessing` and `--jobs 6` this would run 6 processes in parallel. Note that each process can peak at up to 10 GB of RAM (depending on the used classifier and vetoes).

P.6. Computing limits

To compute limits we now feed each of the produced HDF5 output files of the `likelihood` program (in pairs including Run-2 and Run-3 output files) into `mcmc_limit_calculation`. As the number of

combinations can be quite large, this can again be automated using the `Analysis/runLimits` program.

Let's continue with the example from above and pretend we wish to run `1000` toy candidate sets for each input pair, but only those that include the line veto. This can be accomplished by using the `--prefix` argument to `runLimits`, which understands standard glob patterns.

```
./runLimits \
    --path $DATA/classifier/mlp \
    --prefix "lhood_c18_R2_crAll_*line*" \
    --exclude "_septem_" \
    --outpath $DATA/limits/mlp/ \
    --energyMin 0.2 --energyMax 12.0 \
    --axionModel <path_to_differential_axion_flux>.csv \
    --axionImage <path_to_axion_image>.csv \
    --combinedEfficiencyFile <path_to_detection_efficiency>.csv \
    --switchAxes \
    --nmc 1000 \
    --dryRun
```

Note that the CSV files mentioned here are found in the `resources` directory of the repository of this thesis. See the extended thesis on how to produce them.

To produce the observed limit, we run:

```
mcmc_limit_calculation \
    limit \
    -f <path_to_background_run2>.h5 \
    -f <path_to_background_run3>.h5 \
    --tracking <path_to_tracking_run2>.h5 \
    --tracking <path_to_tracking_run3>.h5 \
    --axionModel <path_to_differential_axion_flux>.csv \
    --axionImage <path_to_axion_image>.csv \
    --combinedEfficiencyFile <path_to_detection_efficiency>.csv \
    --switchAxes \
    --path "" \
    --years 2017 --years 2018 \
    --σ_p 0.05 \
    --energyMin 0.2 --energyMax 12.0 \
    --limitKind lkMCMC \
    --outpath ~/phd/Figs/trackingCandidates/ \
    --suffix ""
```

where the `<path_to_*_run>` refers to the output HDF5 file of the `likelihood` program for the best

setup as described in sec. 12.11. `--σ_p` refers to the position uncertainty to use. The signal and background uncertainties have the real value placed as the default.

With this done, you have successfully reproduced the entire final results of the thesis! As mentioned in chapter 2, see the extended thesis for commands, code snippets and more information about how each figure and table is produced exactly.

Q. Average distance X-rays travel in argon at CAST conditions Appendix

Note: This section and its subsections are also available as a standalone document titled `SolarAxionConversionPoint`. It is linked in the extended thesis and this appendix is only a minor modification.

In order to be able to compute the correct distance to use in the raytracer for the position of the axion image, we need a good understanding of where the X-ray will generally convert in the gas.

By combining the expected axion flux (folded with the telescope efficiency and window transmission to get the correct energy distribution) with the absorption length [1] of X-rays at different energies we can compute a weighted mean of all X-rays and come up with a single number. The difficulty lies in combining the statistical process of absorption with the incoming flux distribution. We implement a numerical Monte Carlo approach in literate programming style below.

Q.1. Calculate conversion point numerically

In order to calculate the conversion point, we need:

- random sampling logic
- sampling from exponential distribution depending on energy
- the axion flux, telescope effective area and window absorption

Let's start by importing the modules we need:

```
import helpers / sampling_helper # sampling distributions
import unchained                 # sane units
import ggplotnim                 # see something!
import xrayAttenuation           # window efficiencies
import math, sequtils
```

[1] This was one of the reasons I wrote xrayAttenuation.

where the `sampling_helpers` is a small module to sample from a procedure or a sequence.

In addition let's define some helpers:

```
from os import `/`, expandTilde
const ResourcePath = "~/org/resources".expandTilde
const OutputPath = "~/phd/Figs/axion_conversion_point_sampling/".expandTilde

proc thm(): Theme =
  ## A shorthand to define a `ggplotnim` theme that looks nice
  ## in the thesis
  result = themeLatex(fWidth = 0.9, width = 600, baseTheme = singlePlot)
```

Now let's read the LLNL telescope efficiency as well as the axion flux model. Note that we may wish to calculate the absorption points not only for a specific axion flux model, but potentially any other kind of signal. We'll build in functionality to disable different contributions.

```
let flux = "solar_axion_flux_differential_g_ae_1e-13_g_ag_1e-12_g_aN_1e-15.csv"
let dfAx = readCsv(ResourcePath / flux)
  .filter(f{`type` == "Total flux"})
let llnl = "llnl_xray_telescope_cast_effective_area_parallel_light_DTU_thesis.csv"
let dfLLNL = readCsv(ResourcePath / llnl)
  .mutate(f{"Efficiency" ~ idx("EffectiveArea[cm²]") / (PI * 2.15 * 2.15)})
```

Note: to get the differential axion flux use `readOpacityFile` from `https://github.com/jovoy/AxionElectronLimit`. It generates the CSV file.

Next up we need to define the material properties of the detector window in order to compute its transmission.

```
let Si3N4 = compound((Si, 3), (N, 4)) # actual window
const ρSiN = 3.44.g•cm⁻³
const lSiN = 300.nm                   # window thickness
let Al = Aluminium.init()             # aluminium coating
const ρAl = 2.7.g•cm⁻³
const lAl = 20.nm                     # coating thickness
```

With these numbers we can compute the transmission at an arbitrary energy. In order to compute the correct inputs for the calculation we now have everything. We wish to compute the following, the intensity $I(E)$ is the flux that enters the detector

$$I(E) = f(E) \cdot \varepsilon_{\text{LLNL}} \cdot \varepsilon_{\text{Si}_3\cdot\text{N}_4} \cdot \varepsilon_{\text{Al}}$$

where $f(E)$ is the solar axion flux and the ε_i are the efficiencies associated with the telescope and transmission of the window. The idea is to sample from this intensity distribution to get a realistic set of X-rays as they would be experienced in the experiment. One technical aspect still to be done is an interpolation of the axion flux and LLNL telescope efficiency to evaluate the data at an arbitrary energy as to define a function that yields $I(E)$.

> Important note: We fully neglect here the conversion probability and area of the magnet bore. These (as well as a potential time component) are purely constants and do not affect the **shape** of the distribution $I(E)$. We want to sample from it to get the correct weighting of the different energies, but do not care about absolute numbers. So differential fluxes are fine.

The idea is to define the interpolators and then create a procedure that captures the previously defined properties and interpolators.

```
from numericalnim import newLinear1D, eval
let axInterp = newLinear1D(dfAx["Energy", float].toSeq1D,
                           dfAx["diffFlux", float].toSeq1D)
let llnlInterp = newLinear1D(dfLLNL["Energy[keV]", float].toSeq1D,
                             dfLLNL["Efficiency", float].toSeq1D)
```

With the interpolators defined let's write the implementation for $I(E)$:

```
proc I(E: keV): float =
  ## Compute the intensity of the axion flux after telescope & window eff.
  ##
  ## Axion flux and LLNL efficiency can be disabled by compiling with
  ## `-d:noAxionFlux` and `-d:noLLNL`, respectively.
  result = transmission(Si3N4, ρSiN, lSiN, E) * transmission(Al, ρAl, lAl, E)
  when not defined(noAxionFlux):
    result *= axInterp.eval(E.float)
  when not defined(noLLNL):
    result *= llnlInterp.eval(E.float)

```

Let's test it and see what we get for e.g. 1 keV:

```
echo I(1.keV)
```

yields $1.249e20$. Not the most insightful, but it seems to work. Let's plot it:

```
let energies = linspace(0.01, 10.0, 1000).mapIt(it.keV)
let Is = energies.mapIt(I(it))
block PlotI:
  let df = toDf({ "E [keV]" : energies.mapIt(it.float),
                  "I" : Is })
  ggplot(df, aes("E [keV]", "I")) +
    geom_line() +
    ggtitle("Intensity entering the detector gas") +
    margin(left = 3.0) + thm() +
    ggsave(OutputPath / "intensity_axion_conversion_point_simulation.pdf")
```

shown in fig. Q.1. It looks exactly as we would expect.

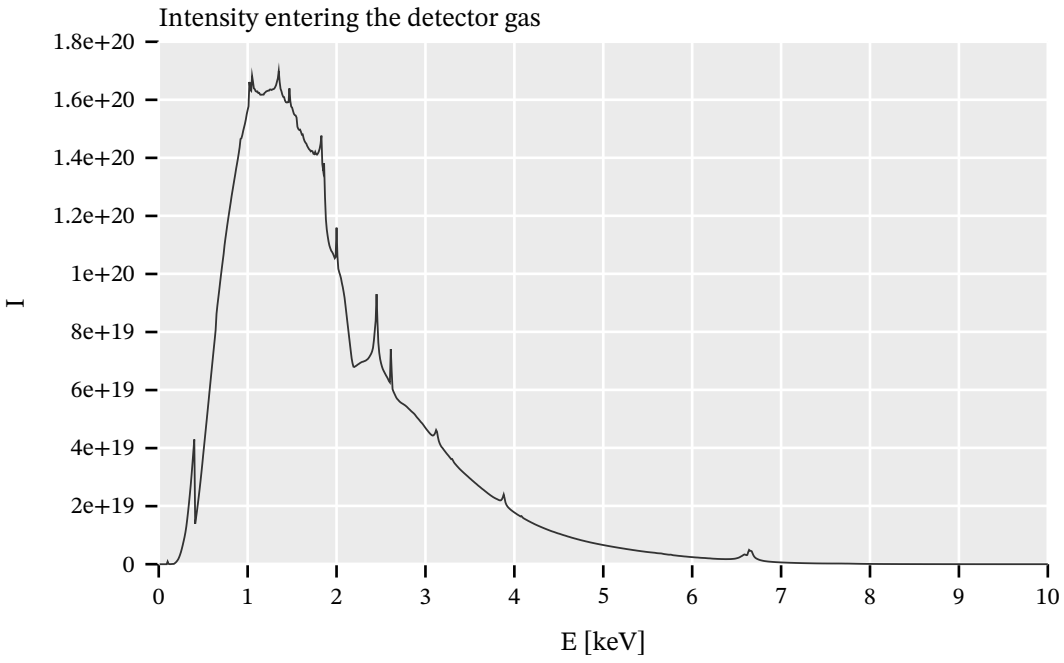


Figure Q.1.: Intensity that enters the detector taking into account LLNL telescope and window efficiencies as well as the solar axion flux

Now we define the sampler for the intensity distribution $I(E)$, which returns an energy weighted by $I(E)$:

```
let Isampler = sampler(
  (proc(x: float): float = I(x.keV)), # wrap `I(E)` to take `float`
  0.01, 10.0, num = 1000 # use 1000 points for EDF & sample in 0.01 to 10 keV
)
```

and define a random number generator:

```
import random
var rnd = initRand(0x42)
```

First we will sample 100,000 energies from the distribution to see if we recover the intensity plot from before.

```
block ISampled:
  const nmc = 100_000
  let df = toDf( {"E [keV]" : toSeq(0 ..< nmc).mapIt(rnd.sample(Isampler)) })
  ggplot(df, aes("E [keV]")) +
    geom_histogram(bins = 200, hdKind = hdOutline) +
    ggtitle("Energies sampled from I(E)") +
    thm() +
    ggsave(OutputPath / "energies_intensity_sampled.pdf")
```

This yields fig. Q.2, which clearly shows the sampling works as intended.

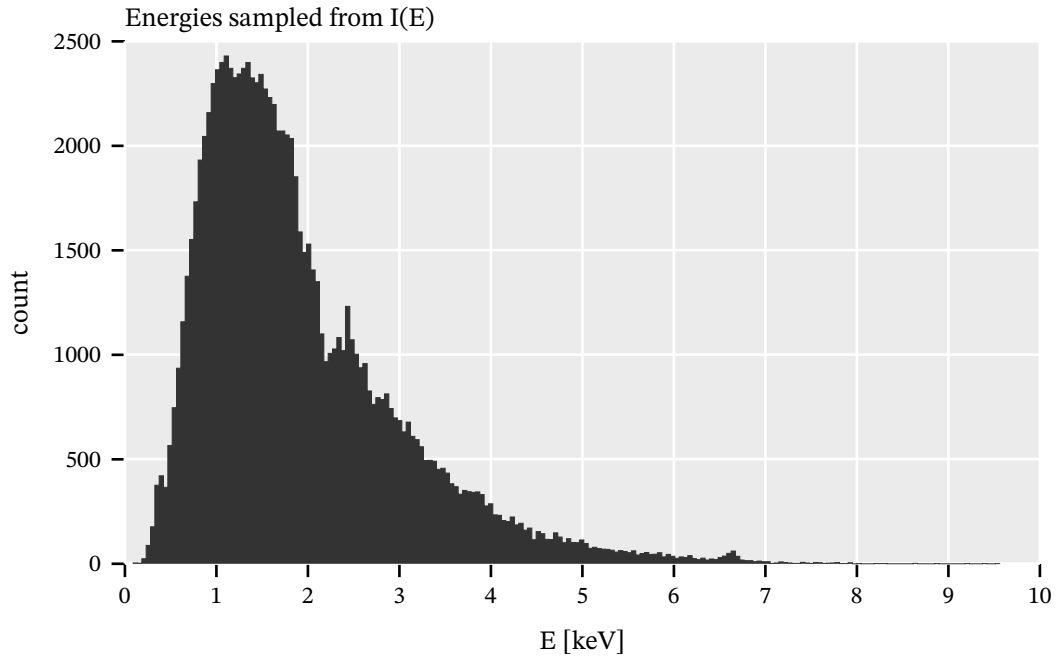


Figure Q.2.: Energies sampled from the distribution $I(E)$ using 100k samples. The shape is nicely reproduced, here plotted using a histogram of 200 bins.

The final piece now is to use the same sampling logic to generate energies according to $I(E)$, which correspond to X-rays of said energy entering the detector. For each of these energies then sample from the Beer-Lambert law (see sec. 5.1.1)

$$I(z) = I_0 \exp\left[-\frac{z}{l_{\text{abs}}}\right],$$

where I_0 is some initial intensity and l_{abs} the absorption length. The absorption length is computed from the gas mixture properties of the gas used at CAST, namely argon/isobutane 97.7/2.3 at 1050 mbar. It is the inverse of the attenuation coefficient μ_M

$$l_{\text{abs}} = \frac{1}{\mu_M}$$

where the attenuation coefficient is computed via

$$\mu_m = \frac{N_A}{M * \sigma_A}$$

with N_A Avogadro's constant, M the molar mass of the compound and σ_A the atomic absorption cross section. The latter again is defined by

$$\sigma_A = 2 r_e \lambda f_2$$

with r_e the classical electron radius, λ the wavelength of the X-ray and f_2 the second scattering factor. Scattering factors are tabulated for different elements, for example by NIST and Henke. For a further discussion of this see the README and implementation of `xrayAttenuation` [195].

We will now go ahead and define the CAST gas mixture:

```
proc initCASTGasMixture(): GasMixture =
  ## Returns the absorption length for the given energy in keV for CAST
  ## gas conditions:
  ## - Argon / Isobutane 97.7 / 2.3 %
  ## - 20°C ( for this difference in temperature barely matters)
  let arC = compound((Ar, 1)) # need Argon gas as a Compound
  let isobutane = compound((C, 4), (H, 10))
  # define the gas mixture
  result = initGasMixture(293.K, 1050.mbar, [(arC, 0.977), (isobutane, 0.023)])
let gm = initCASTGasMixture()
```

To sample from the Beer-Lambert law with a given absorption length we also define a helper that returns a sampler for the target energy using the definition of a normalized exponential distribution

$$f_e(x, \lambda) = \frac{1}{\lambda} \exp\left[-\frac{x}{\lambda}\right].$$

> Note: The sampling of the conversion point is the crucial aspect of this. Naively we might want to sample between the detector volume from 0 to 3 cm. However, this skews our result. Our calculation depends on the energy distribution of the incoming X-rays. If the absorption length is long enough the probability of reaching the readout plane and

thus not being detected is significant. Restricting the sampler to 3 cm would pretend that independent of absorption length we would always convert within the volume, giving too large a weight to the energies that should sometimes not be detected!

Let's define the sampler now. It takes the gas mixture and the target energy. A constant `SampleTo` is defined to adjust the position to which we sample at compile time (to play around with different numbers).

```
proc generateSampler(gm: GasMixture, targetEnergy: keV): Sampler =
  ## Generate the exponential distribution to sample from based on the
  ## given absorption length
  # `xrayAttenuation` `absorptionLength` returns number in meter!
  let λ = absorptionLength(gm, targetEnergy).to(cm)
  let fnSample = (proc(x: float): float =
                    result = expFn(x, λ.float) # expFn = 1/λ · exp(-x/λ)
  )
  const SampleTo {.intdefine.} = 20 ## `SampleTo`, set via `-d:SampleTo=<int>`
  let num = (SampleTo.float / 3.0 * 1000).round.int # # of points to sample at
  result = sampler(fnSample, 0.0, SampleTo, num = num)
```

Note that this is inefficient, because we generate a new sampler from which we only sample a single point, namely the conversion point of that X-ray. If one intended to perform a more complex calculation or wanted to sample orders of magnitude more X-rays, one should either restructure the code (i.e. sample from known energies and then reorder based on the weight defined by $I(E)$) or cache the samplers and pre-bin the energies.

For reference let's compute the absorption length as a function of energy for the CAST gas mixture:

```
block GasAbs:
  let Es = linspace(0.03, 10.0, 1000)
  let lAbs = Es.mapIt(absorptionLength(gm, it.keV).m.to(cm).float)
  let df = toDf({ "E [keV]" : Es,
                  "l_abs [cm]" : lAbs })
  ggplot(df, aes("E [keV]", "l_abs [cm]")) +
    geom_line() +
    ggtitle(r"Absorption length of X-rays in CAST gas mixture: \\" & $gm) +
    margin(top = 1.5) +
    thm() +
    ggsave(OutputPath / "cast_gas_absorption_length.pdf")
```

which yields fig. Q.3

So, finally: let's write the MC sampling!

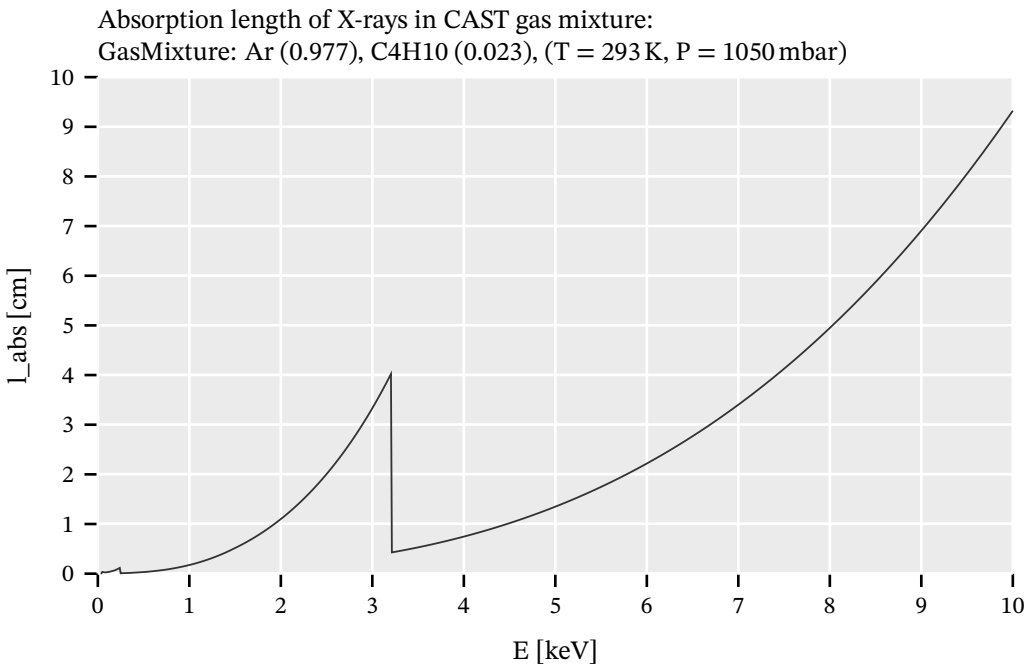


Figure Q.3.: Absorption length in the CAST gas mixture as a function of X-ray energy.

```
const nmc = 500_000 # start with 100k samples
var Es = newSeqOfCap[keV](nmc)
var zs = newSeqOfCap[cm](nmc)
while zs.len < nmc:
  # 1. sample an energy according to `I(E)`
  let E = rnd.sample(Isampler).keV
  # 2. get the sampler for this energy
  let distSampler = generateSampler(gm, E)
  # 3. sample from it
  var z = Inf.cm
  when defined(Equiv3cmSampling): ## To get the same result as directly sampling
                                  ## only up to 3 cm use the following code
    while z > 3.0.cm:
      z = rnd.sample(distSampler).cm
  elif defined(UnboundedVolume): ## This branch pretends the detection volume
                                 ## is unbounded if we sample within 20cm
    z = rnd.sample(distSampler).cm
  else: ## This branch is the physically correct one. If an X-ray reaches the
        ## readout plane it is _not_ recorded, but it was still part of the
        ## incoming flux!
    z = rnd.sample(distSampler).cm
    if z > 3.0.cm: continue # just drop this X-ray
  zs.add z
```

```
  Es.add E
```

Great, now we have sampled the conversion points according to the correct intensity. We can now ask for statistics or create different plots (e.g. conversion point by energies etc.).

```
import stats, seqmath # mean, variance and percentile
let zsF = zs.mapIt(it.float) # for math
echo "Mean conversion position = ", zsF.mean().cm
echo "Median conversion position = ", zsF.percentile(50).cm
echo "Variance of conversion position = ", zsF.variance().cm
```

This prints the following:

```
Mean conversion position = 0.556813 cm
Median conversion position = 0.292802 cm
Variance of conversion position = 0.424726 cm
```

We see the mean conversion position is at about 0.56 cm. If we consider the median it's only 0.29 cm (the number we use in R.4). This number provides the target for the raytracing of the axion image as an offset from the focal length in sec. R.4.

Let's plot the conversion points of all sampled (and recorded!) X-rays as well as what their distribution against energy looks like.

```
let dfZ = toDf({ "E [keV]" : Es.mapIt(it.float),
                 "z [cm]"  : zs.mapIt(it.float) })
ggplot(dfZ, aes("z [cm]")) +
  geom_histogram(bins = 200, hdKind = hdOutline) +
  ggtitle("Conversion points of all sampled X-rays according to I(E)") +
  thm() +
  ggsave(OutputPath / "sampled_axion_conversion_points.pdf")
ggplot(dfZ, aes("E [keV]", "z [cm]")) +
  geom_point(size = 0.5, alpha = 0.2) +
  ggtitle("Conversion points of all sampled X-rays according to I(E) " &
    "against their energy") +
  thm() +
  ggsave(OutputPath / "sampled_axion_conversion_points_vs_energy.pdf",
         dataAsBitmap = true)
```

The former is shown in fig. Q.4. The overlapping exponential distribution is obvious, as one would expect. The same data is shown in fig. Q.5, but in this case not as a histogram, but by their energy as a scatter plot. We can clearly see the impact of the absorption length on the conversion points for each energy!

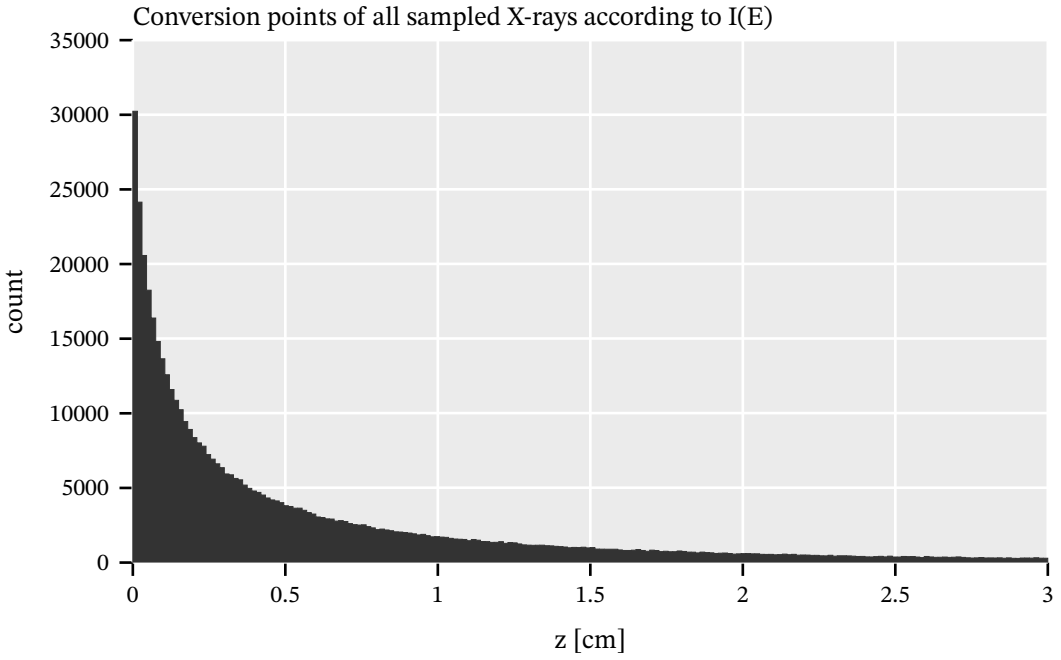


Figure Q.4.: Distribution of the conversion points of all sampled X-rays for which conversion in the detector took place as sampled from $I(E)$.

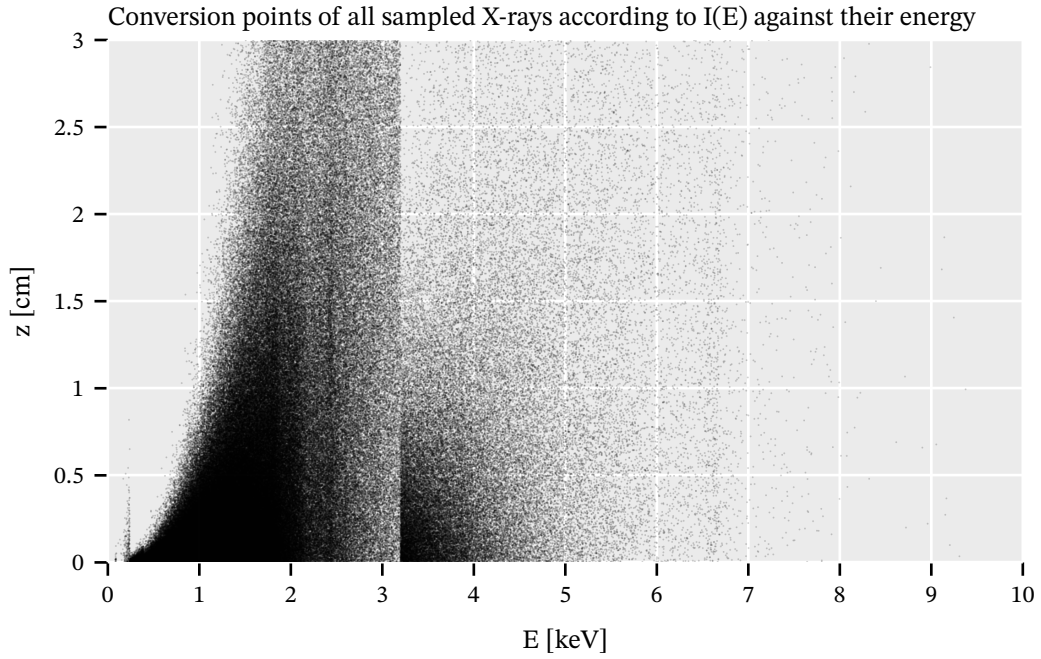


Figure Q.5.: Distribution of the conversion points of all sampled X-rays for which conversion in the detector took place as sampled from $I(E)$ as a scatter plot against the energy for each X-ray.

R. Raytracing Software:Appendix

Contents

In this appendix we will introduce the concept of raytracing, present our raytracer in sec. R.1, give a more technical overview of the LLNL telescope in sec. R.2, show comparisons of our raytracer to PANTER measurements, sec. R.3 and finally compute the axion image expected for CAST behind the LLNL telescope, sec. R.4.

Raytracing is a technique for the rendering of computer-generated images [1]. The classical raytracing algorithm goes back to a paper by Turner Whitted in 1979 [225]. It is essentially a recursive algorithm, which shoots rays ("photons") from a camera into a scene of 3D objects. Interactions with the objects follows geometrical optics. The ray equation

$$\vec{r}(t) = \vec{o} + t\vec{d},$$

describes the ray vector $\vec{r}$, how it propagates from its origin $\vec{o}$ along the direction $\vec{d}$ as a function of the parameter t (it is no coincidence that t evokes the notion of time). Ray-object intersections tests are performed based on the ray equation and parametrizations of scene objects (either analytical parametrizations for geometric primitives like spheres, cones and the like, or complex objects made up of a large number of triangles). Each ray is traced until a certain 'depth', defined by the maximum number of reflections a ray may undergo, is reached and the recursion is stopped.

Building on this more sophisticated methods were invented, in particular the 'path tracing' algorithm and the introduction of the 'rendering equation' [125, 115]. Here the geometrical optics approximation is replaced by the concepts of radiometry (power, irradiance, radiance, spectral radiance and so on) embedding the concept in more robust physical terms. At the heart of it, the rendering

[1] The history of raytracing goes back much further than its application to computer graphics, though.

equation is a statement of conservation of energy. To generate images of a scene, the most interesting quantity is the exitant radiance L_o at a point on a surface. It is the sum of the emitted radiance L_e and the scattered radiance $L_{o,s}$,

$$L_o = L_e + L_{o,s}.$$

In the context of rendering, the emitted radiance L_e is commonly a property of the materials in the scene. $L_{o,s}$ needs to be computed via the scattering equation

$$L_{o,s}(\vec{x}, \omega_o) = \int_{S^2} L_i(\vec{x}, \omega_i) f_s(\vec{x}, \omega_i \mapsto \omega_o) \sin\vartheta \, \mathrm{d} \sin\vartheta \, \mathrm{d}\phi.$$

L_i is the incident radiance at point $\vec{x}$ and direction ω_i (do not confuse it with an energy). f_s is the bidirectional scattering distribution function (BSDF) and describes the scattering properties of a material surface. In the context of an equilibrium in radiance (the light description does not change over time in the scene) the incident radiance can be expressed as

$$L_i(\vec{x}, \omega) = L_o(\mathbf{x}_{\mathcal{M}}(\vec{x}, \omega), -\omega).$$

Here $\mathbf{x}_{\mathcal{M}}(\vec{x}, \omega)$ is the 'ray-casting function', which yields the closest point on the set of surfaces $\mathcal{M}$ of the scene visible from $\vec{x}$ in direction ω. These expressions can be combined into a form independent of L_i to

$$L_o(\vec{x}, \omega_o) = L_e(\vec{x}, \omega_o) + \int_{S^2} L_o(\mathbf{x}_{\mathcal{M}}(\vec{x}, \omega_i), -\omega_i) f_s(\vec{x}, \omega_i \mapsto \omega_o) \sin\vartheta \, \mathrm{d} \sin\vartheta \, \mathrm{d}\phi,$$

which is the 'light transport equation', commonly used nowadays in path tracing based computer graphics.

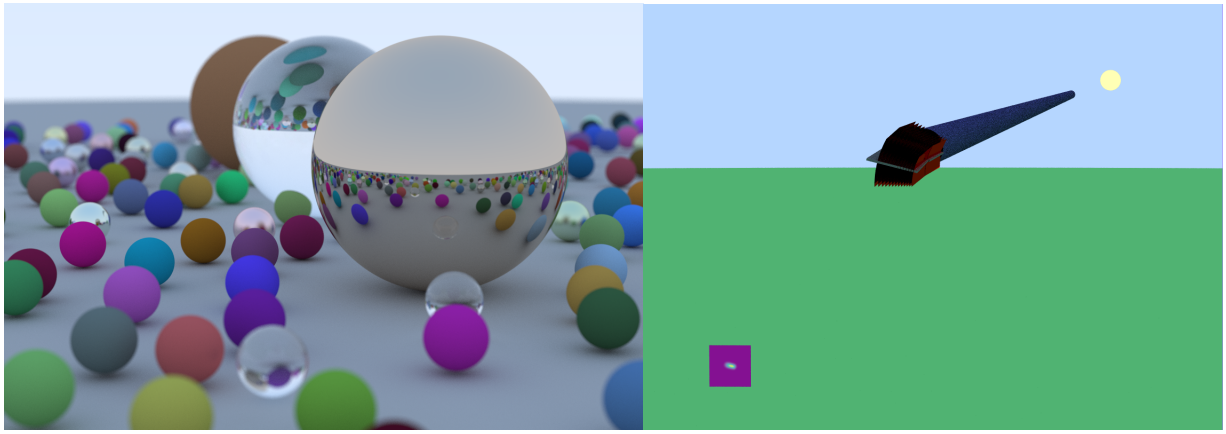
The fundamental problem with raytracing, and path tracing based on the light transport equation in particular, is the extreme computational cost. Monte Carlo based approaches to sample only relevant directions and space partitioning data structures are employed to reduce this cost. One of the seminal documents of modern Monte Carlo based rendering is Eric Veach's PhD thesis [217], which the notation used above is based on. If you are interested in the topic I highly recommend at least a cursory look at it.

Movies have made the transition from rasterization based computer graphics in *Toy Story* (1995) over the course of 15 to 20 years [2] to full path tracing graphics commonly used today. The use of offline rendering utilizing many compute hours per frame of the final movie allowed movies to introduce path tracing at the end of the 90s (*Bunny*, an animated short released in 1998). In recent years the idea of real time path tracing has become tangible, in big parts due to the – at the time bold – bet

[2] The first feature length movie entirely rendered using path tracing was *Monster House* (2006).

by Nvidia. They dedicated specific parts of their 'Turing' architecture of graphics processing units (GPUs) in 2018 to accelerate raytracing. Starting with path traced versions of old video games, like *Quake II RTX* running in real time on 'Turing' GPUs, in 2023 we have path traced versions of modern high budget ("AAA") video games like *Cyberpunk 2077* and *Alan Wake 2* [3]. This is in parts due to the general increase in compute performance, partially due to specific hardware features and a large part due to usage of machine learning (to denoise and upsample images and further reconstruct missing rays using ML). For an overview of the developments in the movie industry to its adoption of path tracing, have a look at [61]. To understand how path tracing is implemented in video games on modern GPUs, see the 'Raytracing Gems' series [100, 4].

R.1. TrAXer - An interactive axion raytracer



(a) RTOW spheres example (b) CAST setup

Figure R.1.: (a) is the reproduction of the final result of the book 'Ray Tracing in One Weekend' (RTOW) [205] I originally followed. (b) is a very narrow angle of view on the CAST setup. In the bottom left is the image sensor with the size of a GridPix. In the middle with a reddish color is the LLNL telescope sitting in front of the CAST magnet bore. On the right is the Sun. Note that only the essential parts are simulated floating in the air. The angle and view from above makes the image sensor look a bit odd.

The simulation of light based on physical principles as done in path tracing algorithms combined with the performance of modern computers make it an interesting solution to light propagation problems, which are difficult if not impossible to solve numerically. Calculation of the expected image produced by the X-ray telescope for a realistic X-ray telescope and solar axion emission is one such problem. In [134] Christoph Krieger already wrote a basic hardcoded [4] raytracer for the 2014/15 data taking behind the ABRIXAS telescope. He approximated the ABRIXAS telescope as a simple lens.

[3] There were of course previous experiments in interactive real time raytracing.

[4] By 'hardcoded' in this context I mean that the scene geometry is entirely embedded in imperative code. Each reflection is calculated manually. Thus, the raytracer is not generic.

For the data taking with the detector of this thesis behind the LLNL telescope, a more precise solution was desired. As such Johanna von Oy wrote a new raytracer for this purpose as part of her master thesis [158]. However, although this code [159] correctly simulates the LLNL (and as a matter of fact also other telescopes) based on their correct layer descriptions, it still uses a similar hardcoded [4] approach, making it inflexible for different scenes. More importantly it makes understanding and verifying the code for correctness very difficult.

What started mainly as a toy project I wrote out of curiosity following the popular 'Ray Tracing in One Weekend' (RTOW) [205] book [5], anyhow made me appreciate the advantage of a generic raytracer (see fig. R.1(a) for the final result of that book, created by this project). Generic meaning that the scene to be rendered is described by objects placed in simulated 3D world space. This reduces the amount of code that needs to be checked for correctness by orders of magnitude (reflection for one material type is a few line function, instead of having lines of code for every single reflection in a hardcoded raytracer). It also makes the program much more flexible to support different setups. In particular though, the concept of a camera independent of the scene geometry allows to visualize the scene (obviously the entire purpose of a normal raytracer!).

As a personal addition to the RTOW project, I implemented rendering to an SDL2 [199] window and handling mouse and keyboard inputs to produce an interactive real-time raytracer (this was early 2021). This sparked the idea to use the same approach for our axion raytracer. I pitched the idea to Johanna, but did not manage to convince her [6]. While working on finalizing the axion image and X-ray finger simulations for this thesis, several big discrepancies between our raytracing results and those used for the CAST Nature paper in 2017 [64] were noticed. The latter were done by Michael Pivovaroff at LLNL using a private in-house raytracer [7]. The need to better verify the correctness of our own raytracing simulation lead me down the path of finishing the project of turning the RTOW based raytracer into an axion X-ray raytracer. This way verifying correctness was easier. Some additional features of the second book of the ROTW series were added (light sources among others) and other aspects were inspired by Physically Based Rendering [170] (in particular the idea of propagating multiple energies in each ray).

The result is `TrAXer` [193], an interactive real-time visible light and X-ray raytracer. In essence it is two raytracers in one. First, a 'normal' visible light raytracer, which the camera uses. Secondly, an X-ray raytracer. Both use the same raytracing logic, but they differ in sampling. For visible light we emit rays from the camera into the scene, while for X-rays we emit from an X-ray source (an object

[5] The nice thing about the first book is that it can literally be implemented in a single weekend!

[6] Some people may just be better at judging which efforts are worth it, than I am, haha! On a serious note, given her other obligations and this not being an urgent problem, we did not pursue it at the time.

[7] At least that is my understanding. There are two public X-ray raytracing packages I am aware of, `McXtrace` [130] and `MT_RAYOR` [156]. There is a chance `MT_RAYOR` may have been used, because it is written in an interpreted programming language, 'Yorick' [232], developed at LLNL. And it was used for NuSTAR raytracing simulations in the past [53].

placed in the scene, for example the Sun). Both of these run in parallel. Scenes are defined by placing geometric primitives with different materials (glass, metal etc.) into the scene.

The X-ray raytracer is used by choosing X-ray emitting materials and adding another object as an X-ray target (another material), which the X-ray sampling will be biased towards. There is further a specific X-ray material an object can be made of, which behaves as expected when hit by an X-ray, calculated using [195]. Behavior is described by the reflectivity computed via the Fresnel equations as discussed in sec. 5.1.2. To simulate reflectivity of the LLNL telescope, the depth graded multilayers of the different layers are taken into account. X-ray transmission is currently not implemented, but only because there is no need for us. The addition would be easy based on mass attenuation coefficients and the Fresnel equations for transmission.

In order to detect and read out results from the X-ray raytracing, we place ”image sensors” into the scene. They use a special material, which accumulates X-rays hitting them, split into spatial pixels. For visible light they simply emit the current values stored in each pixel of the image sensor, mapped to the Viridis color scale.

These things are visible in the example screenshot showing the ”CAST” scene in fig. R.1(b). The view is seen from behind and above the setup with a narrow field of view (like a telephoto lens). Only relevant pieces of the setup are placed into the scene (partially for performance, although bounding volume hierarchies help there, mostly for simplicity). The Sun is visible on the right side, emitting a yellow hue for the visible rays and an invisible X-ray spectrum following the expected solar axion flux (with correct radial emission). The long blueish tube is the inner bore of the CAST magnet. Then, with red hues (in visible light) we have the LLNL telescope. In the bottom left is the image sensor, which has the same physical size as the center GridPix of the Septemboard detector and is placed in the focal point of the telescope. As every object of the scene is floating above the earth, combined with the telephoto like view, the sensor looks a bit odd. [8]

R.2. A few more details about the LLNL telescope

The Lawrence Livermore National Laboratory, introduced in sec. 4.1.2, is the X-ray telescope of interest for us. Let us now first get an overview of the telescope and its construction itself, before we move on to comparing the TrAXer simulations with PANTER measurements and raytracing simulations from Michael Pivovaroff from LLNL.

Its design follows the technology developed for the NuSTAR mission [102, 101, 131, 67, 99]. This means it is not a true Wolter optic (meaning parabolic and hyperbolic mirrors), but instead uses a

[8] You may wonder why there is nothing shielding the image sensor from direct illumination by the Sun. The reason is that the X-rays emitted from the Sun are only sampled into the entrance of the magnet bore. This way we save compute and do not need a mask to shield the sensor.

cone approximation [169], which are easier to produce and in particular for CAST provide more than enough angular resolution.

It consists of two sets of 13 layers. Each mirror has a physical length of 225 mm and thickness of $d_{\text{glass}} = 0.21\,\text{mm}$. These mirrors are made of glass, which are thermally-formed ("slumped") into the conical shapes. Production of the mirrors happened at the National Space Institute at the Technical University of Denmark (DTU Space) as part of the PhD thesis of Anders Jakobsen [120]. The first sets of layers i describe truncated coni with a remaining height of 225 mm, an arc angle of 30 °, a radius on the opening end [9] of $R_{1,i}$ and a cone angle of α_i. That is for each layer a non-truncated cone would have a total height of $h = R_{1,i}/\tan\alpha_i$. The second set of 13 layers only differs by their radii, $R_{4,i}$ on the opening end and an angle of $3\alpha_i$. In the horizontal direction along the optical axis of the telescope, there is a spacing of $x_{\text{sep}} = 4\,\text{mm}$ between each set of layers. Due to the tilted angle the physical distance between each set is minutely larger. Combined the telescope thus has a length of roughly 454 mm.

One can define additional helpful radii, $R_{2,i}$ the minimum radius of the first set of cones, $R_{4,i}$ the maximum radii of the opening end of the second set of mirrors and finally $R_{3,i}$, the radius at the mid point where the hypothetical extension of the first layer and last layer would touch at $x = 2\,\text{mm}$ from each set of mirrors. Equations R.1 specify how these numbers are related with l the length of the mirrors.

$$R_{2,i} = R_{3,i} + 0.5 x_{\text{sep}} \tan(\alpha_i) \tag{R.1}$$

$$R_{1,i} = R_{2,i} + l \sin(\alpha_i) \tag{R.2}$$

$$R_{4,i} = R_{3,i} - 0.5 x_{\text{sep}} \tan(3\alpha_i) \tag{R.3}$$

$$R_{5,i} = R_{4,i} - l \sin(3\alpha_i) \tag{R.4}$$

which we can rewrite to compute everything from $R_{1,i}$ by:

$$\begin{aligned} R_{2,i} &= R_{1,i} - l \sin(\alpha_i) \\ R_{3,i} &= R_{2,i} - 0.5 x_{\text{sep}} \tan(\alpha_i) \\ R_{4,i} &= R_{3,i} - 0.5 x_{\text{sep}} \tan(3\alpha_i) \\ R_{5,i} &= R_{4,i} - l \sin(3\alpha_i) \end{aligned}$$

I emphasize this, because this conical nature describes the *entire telescope* as long as the radii $R_{1,i}$, $R_{4,i}$ and angle α_i are known. [10]

[9] By 'opening end' I talk about the side of a truncated cone with the larger radius. For an ice cream cone it would simply be the radius at the top.

[10] An alternative, equivalent description replaces the angle by the minimum radii of the coni.

Alternatively, this can also be used to *construct* an optimal conical telescope from a starting radius $R_{1,0}$ (typically called the "mandrel"), if the iterative condition $R_{3,i+1} = R_{1,i} + d_{\text{glass}}$ is employed (in this case some target parameters are needed of course, like focal length, x_{sep} and a few more).

To calculate the focal length f of an X-ray telescope of the Wolter type, we can use the Wolter equation,

$$\tan(4\alpha_i) = \frac{R_{3,i,\text{virtual}}}{f}$$

where α_i is the angle of the i^{th} layer and $R_{3,i,\text{virtual}}$ is the radius corresponding to the virtual reflection point. Due to the double reflection inherent to a Wolter telescope, the incoming ray picks up a total angle of 4α (0 to 2α after first reflection of the shell with angle α, then 2α to 4α after reflecting from second shell of angle 3α). The virtual radius is the radius at the point if one extends the incoming ray through the first set of mirrors and reflects at a single virtual mirror of an angle 4α precisely at the midpoint of the telescope. This is illustrated in fig. R.2, which is an adapted version of fig. 4.6 from A. Jakobsen's PhD thesis [120] to better illustrate this.

In addition to the PhD thesis mentioned, there is a paper [28] about the telescope for CAST and initial results. **Note though**, that *both* the PhD thesis and the paper contain contradicting and wrong information about the details of the telescope design. Neither (nor combined do they) describe the real telescope. [11] , [12] Thanks to the help of and personal communication with Jaime Ruz and Julia Vogel we were able to both clear up confusions and find the original numbers of the telescope that were actually built. Table R.1 gives an overview of the telescope design. It is adapted from [28], but modified to use the correct numbers. Then in tab. R.2 is the list of all layers and their corresponding radii in detail as given to me by Jaime Ruz.

Having discussed the physical *layout* of the telescope, let us quickly talk about the telescope coatings. In contrast to telescope like the ABRIXAS or XMM-Newton optics, which use a single layer gold coating, the NuSTAR design uses a depth graded multilayer coating as introduced in sec. 5.1.2. There are four different 'recipes', used for different layers. All recipes are a depth graded multilayer of different number of Pt/C layers.

> Note that in this terminology the high Z material (platinum) is actually *below* the low Z material (carbon) contrary to what Pt/C might imply!

Table R.3 gives an overview of which recipes are used for which layer and how they differ. d_{min} is

[11] Anders' thesis contains numbers for a telescope design with a focal length of 1530 mm instead of 1500 mm, possibly due to a typo in a form of the Wolter equation. The paper contains schematics with factually wrong annotations and more wrong (or outdated?) numbers.

[12] Let me be very clear: by no means I am trying to demean Anders work! These things happen. It is just a shame to not have any public, accurate information about the telescope. In particular the paper [28] should have received an errata.

Property	Value
Mirror substrates	glass, Schott D263
Substrate thickness	0.21 mm
l, length of upper and lower mirrors	225 mm
Overall telescope length	$\sim$ 454 mm
f , focal length	1500 mm
Layers	13
Total number of individual mirrors in optic	26
$R_{1,i}$, range of maximum radii	63.24 to 102.38 mm
$R_{3,i}$, range of mid-point radii	62.07 to 100.5 mm
$R_{5,i}$, range of minimum radii	53.88 to 87.19 mm
α, range of graze angles	0.592 to 0.958 °
Azimuthal extent	$\sim$ 30 °

Table R.1.: Properties of the LLNL telescope. Adapted from [28] and modified to match the numbers actually built.

i	R_1 [mm]	R_2 [mm]	R_3 [mm]	R_4 [mm]	R_5 [mm]	α [°]	3α [°]
1	63.2384	60.9121	60.8914	60.8632	53.8823	0.5924	1.7780
2	65.8700	63.4470	63.4255	63.3197	56.0483	0.6170	1.8520
3	68.6059	66.0824	66.0600	65.9637	58.3908	0.6426	1.9288
4	71.4175	68.7898	68.7664	68.6794	60.7934	0.6692	2.0086
5	74.4006	71.6647	71.6404	71.5582	63.3473	0.6967	2.0913
6	77.4496	74.6014	74.5761	74.4997	65.9515	0.7253	2.1773
7	80.6099	77.6452	77.6188	77.5496	68.6513	0.7550	2.2665
8	83.9198	80.8341	80.8067	80.7305	71.4688	0.7858	2.3591
9	87.3402	84.1290	84.1005	84.0137	74.3748	0.8178	2.4553
10	90.8910	87.5495	87.5198	87.4316	77.4012	0.8510	2.5551
11	94.5780	91.1013	91.0704	90.9865	80.5497	0.8850	2.6587
12	98.3908	94.7737	94.7415	94.6549	83.7962	0.9211	2.7662
13	102.381	98.6187	98.5853	98.4879	87.1914	0.9581	2.8778

Table R.2.: Overview of the relevant numbers as the telescope was actually built. Based on a data file sent to me by Jaime Ruz. The values for R_3 and the angles α, 3α were calculated from the values for R_1, R_2 and R_4, R_5 based on equations R.1.

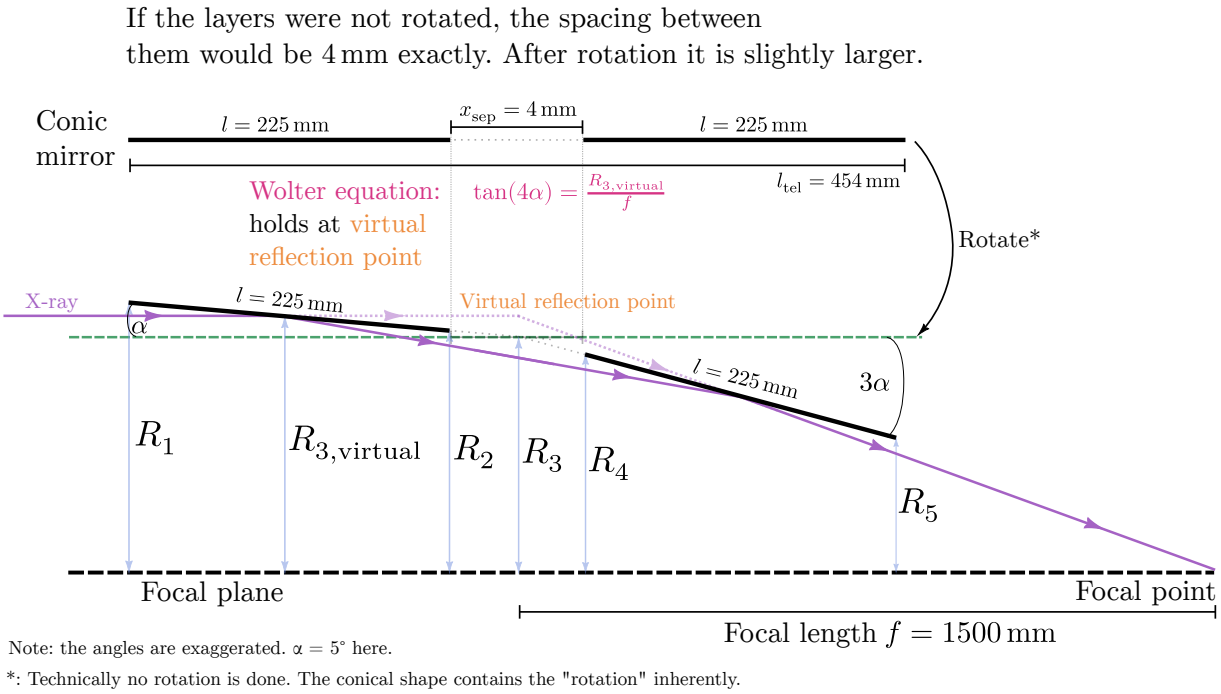


Figure R.2.: Schematic of the reflection of an X-ray on a single layer of the telescope. Angles are exaggerated ($\alpha = 5\,°$ here). The virtual reflection point is the relevant point to consider when calculating the focal length.

the minimum thickness of one multilayer and d_{max} the maximum thickness. The top most multilayer is the thickest. Γ is the ratio of the top to bottom material in each multilayer.

For example recipe 1 has a top multilayer of carbon of a thickness $\Gamma \cdot d_{\mathrm{max}} = 10.125\,\mathrm{nm}$ on top of $(1 - \Gamma) \cdot d_{\mathrm{max}} = 12.375\,\mathrm{nm}$ of platinum. Because recipe 1 only has 2 such multilayers, the bottom Pt/C layer has a combined thickness of d_{min}.

To reiterate the equations that describe the layer thicknesses for N layers as given in sec. 5.1.2, a depth-graded multilayer is described by the equation:

$$d_i = \frac{a}{(b+i)^c} \tag{R.5}$$

where d_i is the depth of layer i (out of N layers),

$$a = d_{\mathrm{min}}(b+N)^c$$

and

$$b = \frac{1 - Nk}{k - 1}$$

with

$$k = \left(\frac{d_{\text{min}}}{d_{\text{max}}}\right)^{\frac{1}{c}}.$$

In all four recipes used for the LLNL telescope the parameter c is set to 1.

Recipe	# of layers	d_{min} [nm]	d_{max} [nm]	Γ
1	2	11.5	22.5	0.45
2	4	7.0	19.0	0.45
3	4	5.5	16.0	0.4
4	2	5.0	14.0	0.4

Table R.3.: Overview of the different depth graded multilayer coatings used for the table. Adapted from fig. 4.11 of [120] with the correct numbers finally built. Parameter c of equation R.5 is set to 1 in all layers.

Calculation of the reflectivity follows the equations also mentioned in the theory section. The reflectivities of each layer are calculated with [195], which is used as part of the material description in `TrAXer`.

R.3. Comparison of TrAXer results with PANTER measurements

The LLNL telescope was tested and characterized at the PANTER X-ray test facility in Munich in July 2016. It was reported on these tests during the 62nd CAST collaboration meeting (CCM) on *<2016-09-26 Mon>* and again in the CCM on *<2017-01-23 Mon>*. [13] These results are valuable to us, because they can be used as a cross-check to verify our `TrAXer` based raytracing simulations.

At the PANTER facility the X-ray source sits 130.297 m away from the center of the telescope (the plane of the virtual reflection). It can be approximated as a source of radius 0.42 mm. The detector used to measure at PANTER is installed on the optical axis (i.e. directly 'in front of' the source). The telescope itself is offset by the radius of its shells, such that its optical axis aligns with the optical axis defined by the X-ray beam (instead of aligning the telescope entrance with the beam axis and the detector offset from it!). In this way the light entering the telescope does not enter perpendicular to the opening, but under an angle.

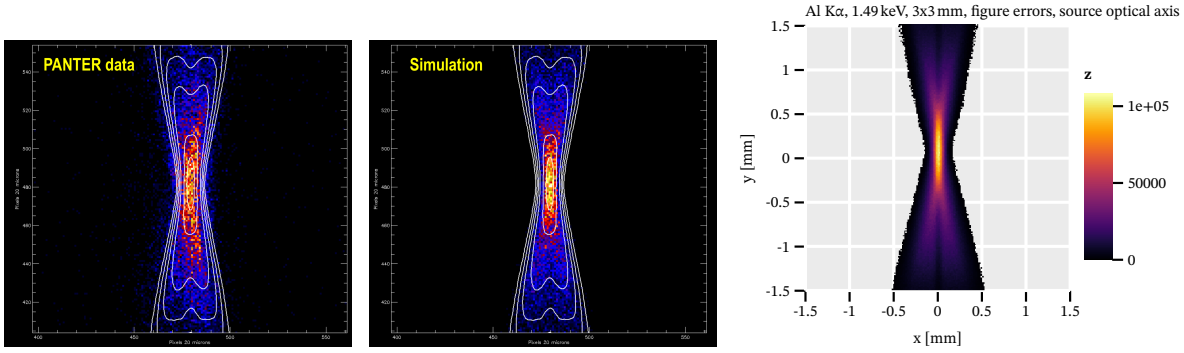
The setup of having such an effective point source implies that the best focal point will not be in the physical focal spot defined by the focal length, but slightly behind. At PANTER this was measured to be at 1519 mm instead of 1500 mm and provides a first check of our raytracer. When reproducing the same setup in TrAXer we find the smallest, most symmetric spot around the 1519 to 1520 mm mark

[13] I mention the exact dates, because if you have access to the indico page of all the CAST collaboration meetings, this should make it easy to find the slides.

as well. [14]

In this setup three different X-ray fluorescence lines were measured. Al Kα (1.49 keV), Ti Kα (4.51 keV) and Fe Kα (6.41 keV). We can compare the images between measurements and simulations as well as compute the 'half power diameter' (HPD). The HPD is the radius around the image center (based on the weighted mean center) in which 50 % of the flux is contained. This is computed based on the encircled energy function (EEF): integrate the flux in radial slices around the center, compute a radial cumulative distribution function and find the radius corresponding to 50 %.

Let's consider the aluminum Kα lines. Fig. R.3 compares the PANTER data, with the LLNL raytracing result and our raytracing result. [15] All three plots show the inner $3 \cdot 3\,\text{mm}^2$ of the image. Generally, the agreement is very good. The "bow tie" shape can be seen in all three figures.



(a) PANTER data (b) LLNL simulation (c) TrAXer

Figure R.3.: (a): Data of the telescope taken at PANTER. (b): Raytracing simulation performed by M. Pivovaroff at LLNL. Both of these are taken from the CCM in Jan 2017, slide 11. (c): Simulation performed with TrAXer. All three plots show the inner $3 \cdot 3\,\text{mm}^2$ of the image.

Computing the encircled energy function yields a figure as shown in fig. R.4. We see the HPD shown as the red line at a radius of $r_{\text{HPD}} \approx 0.79\,\text{mm}$. This is converted to an HPD in arc seconds via

$$\alpha_{\text{HPD}} = \arctan\left(\frac{r_{\text{HPD}}}{f}\right)$$

where f is the focal length of the optic. This leads to an HPD of 216.38″, which is slightly above both the LLNL raytracing simulation (PANTER model) and the PANTER data. Doing this for all three fluorescence lines, we get the numbers shown in tab. R.4, where the differences are even smaller. Therefore, our implementation of the raytracer including figure errors yields very compatible results to the LLNL model developed for PANTER. The usage of TrAXer as the tool of choice for the calculation of the axion image seems justified.

[14] The differences on the order of 1 mm are very small. I haven't implemented an optimization routine to find the absolute smallest value precisely. This is by eye.

[15] I don't show a screenshot of the scene, because the source is obviously tiny and far away. The telescope and sensor are the same as in the CAST scene shown earlier. The CAST magnet bore is not included.

See sec. R.3.1 below for more details on how the figure error is introduced and its parameters determined.

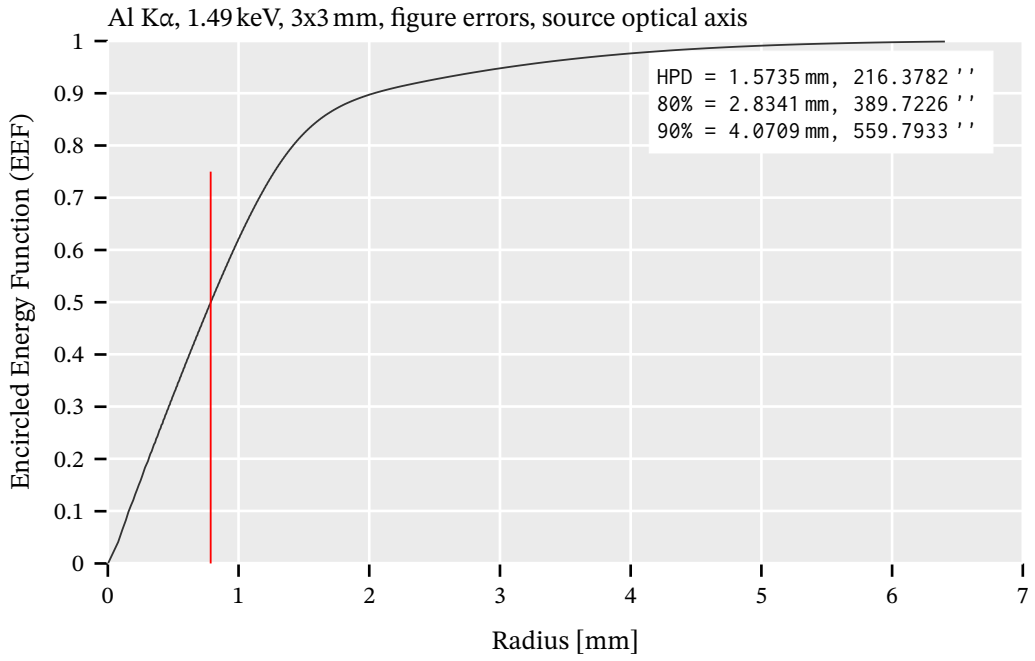


Figure R.4.: Encircled energy function of the aluminum Kα line, corresponding to the data of fig. R.3(c). The HPD comes out to 186.45″.

R.3.1. Definition of the figure error and parameter determination

The figure error implementation has multiple free parameters. Generally speaking upon reflection on a mirror surface, the reflected direction is slightly modified. Let $\vec{r}_i$ be the incoming ray, $\vec{n}$ the surface normal on which the ray reflects and $\vec{r}_r$ the reflected ray. Reflection happens in the plane defined by $\vec{n}$ and $\vec{r}_{i,r}$. For a realistic figure error we wish to mainly vary the reflected direction in this plane around $\vec{r}_r$, but still allow for smaller amounts in directions orthogonal. The vector orthogonal to this plane is

$$\vec{n}_\perp = \vec{n} \times \vec{r}_r.$$

This orthogonal vector $\vec{n}_\perp$ allow us further to construct a vector that is orthogonal to $\vec{r}_r$ but in the plane of the incoming and outgoing ray,

$$\vec{s} = \vec{n}_\perp \times \vec{r}_r.$$

Based on two fuzzing parameters $f_\parallel$ and $f_\perp$ we can then define the real reflected ray to be:

Al Kα (1.49 keV)			
	50% (HPD)	80% circle	90% circle
Point source (perfect mirrors)	168 arcsec	270 arcsec	313 arcsec
Point source (figure errors)	206	387	568
PANTER data	206	397	549
PANTER model	211	391	559
TrAXer (perfect mirrors)	183.19	304.61	351.54
TrAXer (figure errors)	216.38	389.72	559.79
Ti Kα (4.51 keV)			
	50% (HPD)	80% circle	90% circle
Point source (perfect mirrors)	161	259	301
Point source (figure errors)	202	382	566
PANTER data	196	380	511
PANTER model	206	380	559
TrAXer (perfect mirrors)	174.84	288.54	333.75
TrAXer (figure errors)	207.43	378.16	552.55
Fe Kα (6.41 keV)			
	50% (HPD)	80% circle	90% circle
Point source (perfect mirrors)	144	233	265
Point source (figure errors)	184	350	541
PANTER data	196	364	483
PANTER model	185	348	516
TrAXer (perfect mirrors)	160.38	257.79	296.79
TrAXer (figure errors)	189.75	345.20	518.51

Table R.4.: Measurements and simulation results given on slides 18-20 of the CCM slides Jan 2017. 'Point source' refers to LLNL raytracing simulations. PANTER model is an LLNL raytracing model adjusted to best match the PANTER data. Our raytracing results are the TrAXer rows. TrAXer generally overestimates the HPD by about 20″ compared to the perfectc mirror LLNL raytracing results.

$$\vec{r}_{\text{fuzzed}} = \vec{p}_{\text{hit}} + \vec{r}_r + f_{\parallel} \cdot \vec{s} + f_{\perp} \cdot \vec{n}_{\perp}$$

where $\vec{p}_{\text{hit}}$ is the point at which the ray hit a mirror shell. Generally $f_{\parallel}$ and $f_{\perp}$ are parameters sampled from a normal distribution. In practice only $f_{\perp}$ is directly sampled from a single normal distribution, while $f_{\parallel}$ is – effectively – a mix of a narrow and a wide normal distribution. In total there are 5 input parameters, which define the final figure error model.

In order to find the best matching parameters, we perform non-linear optimization using a global derivative-free algorithm [16]. This optimization calls the raytracer with a set of parameters in batch mode [17] up to a fixed set of rays, computes the HPD and radii containing 80 % and 90 % of the flux and finally computes the mean squared error of the target parameters given and the current values.

Ideally of course, one would use a map of the real surface roughness of each mirror shell (as done in [53] for the NuSTAR telescope using `MT_RAYOR`) and use that as the basis for more accurate reflection on the surface.

See fig. R.5 for a comparison of the image without any figure error (assuming perfect mirrors) and the final figure errors (which are the ones used in the table and plots of the previous section). The right plot shows the same data as seen in fig. R.3(c) above. The difference is massive in the sharpness of the data.

R.4. Computing an axion image with TrAXer

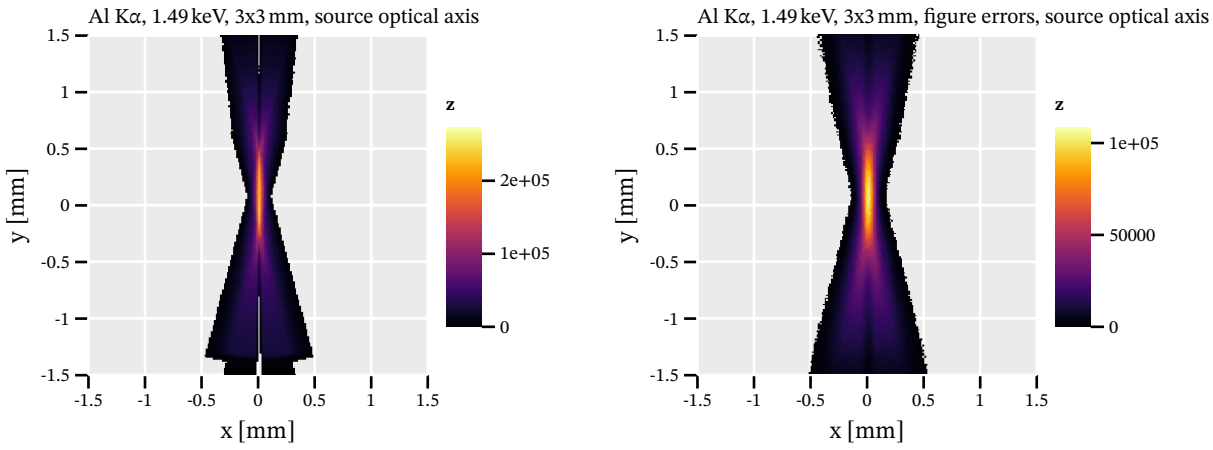
The previous section provides us with reasonable guarantees that our raytracer produces compatible results to PANTER measurements and the LLNL raytracing results. In particular for the axion-electron coupling we do not wish to reuse the raytracing prediction for the solar axion image computed by Michael Pivovaroff at LLNL, as used in the CAST Nature 2017 paper [64] paper. [18] This is because of the different radial emission profile (compare fig. 3.3(b)). And for the chameleon, due to its production in the solar tachocline, this is of course absolutely mandatory.

To simulate the solar axion image, we use the scene already shown in fig. R.1(b). The Sun is placed

[16] To be precise the `NLOPT_GN_DIRECT_L` algorithm of the NLopt [122] library, an implementation of the locally biased 'DIviding RECTangles' algorithm [123, 87]. The algorithm performs a hyperrectangle search of the bounded parameter space. Note that many other algorithms would work fine too, but we want a global search algorithm, as there are possibly different local minima in how the parameters are chosen. Derivative-free is an advantage, as each iteration is expensive, having to run the entire raytracer to accumulate enough statistics ($\mathcal{O}(10\,\text{s})$).

[17] Batch mode just refers to only running the X-ray raytracer without interactivity, graphical interface or the visible light raytracer running.

[18] Let alone the fact that we only got access to these raytracing results mid 2023, but our raytracing work started years before that.



(a) Perfect mirrors

(b) Imperfect mirrors

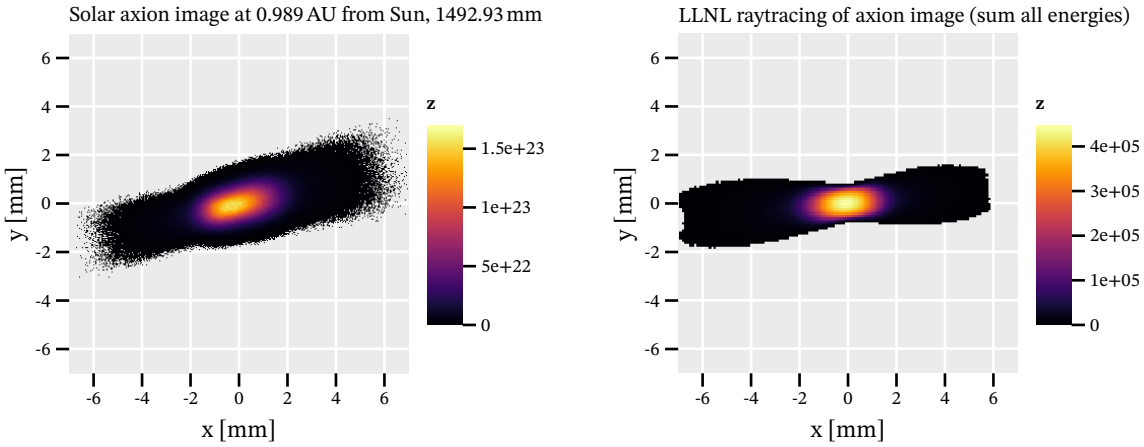
Figure R.5.: (a): TrAXer simulation for Al Kα without any figure error. The resulting image is extremely well defined. (b): TrAXer simulation of Al Kα including a figure error. This is the same image as shown in (c).

as an X-ray emitter [19] at 0.989 AU away [20] from the telescope. It emits X-rays of energies and from radii corresponding to the distribution seen in fig. 3.3(a) calculated using the opacity calculation code [159] also developed by Johanna von Oy during her master thesis [158]. The calculations are based on the AGSS09 solar model [25, 203]. The telescope is rotated under the angle it was installed at CAST, as deduced from the X-ray finger taken at CAST, fig. 9.2(a) (about 14 °). The image sensor is placed slightly in front of the focal point. 1492.93 mm away from the center of the telescope (to the virtual reflection point) instead of 1500 mm to account for the median conversion point of the X-rays of the expected axion electron X-ray flux (as mentioned in sec. 12.8.5 and calculated in Q).

With all these in place, the axion image comes out as seen in fig. R.6(a). In comparison fig. R.6(b) shows the axion image as computed by the LLNL raytracing code (albeit only for the axion-photon coupling and related emission). The similarity between the two images is implying our raytracing simulation is sensible. Small differences are expected, not only because of a different solar emission. The different rotation angle is due to a different assumed rotation of the telescope. Our raytracing image shows a larger bulge in the center at very low intensity. The absolute size of the major flux region is very comparable. The slight asymmetry in our image is due to it being effectively 8 mm in front of the focal point.

[19] The axion-photon conversion probability is handled in the limit code.

[20] This was the average distance to the Sun during our CAST data taking campaign, as discussed in sec. 12.8.2.



(a) TrAXer (b) LLNL

Figure R.6.: (a): TrAXer simulation of the solar axion image using the correct Sun-Earth distance with the image sensor placed 1492.93 mm away from the telescope center. (b): Corresponding raytracing simulation done at LLNL, however for axion-photon production only.

S. Acknowledgments Part5

If you come here after having read (parts of?) the thesis, then first of all my gratitude to you, the reader, for having spent time with my writing. It means a lot.

Of course, without Klaus Desch I wouldn't even have been able to work on this thesis. Thanks a lot, Klaus. And especially thank you for taking the time to do our personal meetings these last years. I really appreciate it!

Next, I want to thank everyone in our group. Although, especially since COVID hit I haven't been to the office much, it was always a great environment to work in. Most notably I want to thank Tobias (Schiffer) and Markus (Gruber) as my longest office mates. We had a lot of fun together (maybe some days a bit too much, hehe) and I hope we can continue even when I'm not around anymore! Johanna (von Oy), you came later, but that doesn't mean I didn't enjoy working with you. On the contrary, it was great. Thanks to being open minded about my software ideas. :)

Some people in our group left a while ago, but you're not forgotten. Christoph (Krieger), thank you for your supervision all those years ago. I hope you are satisfied with my work and maybe a little bit proud! Lucian (Scharenberg), you technically didn't really 'leave' until recently, but we didn't see each other much with you being at CERN for your PhD. Also thank you for the good times and we'll keep in touch! Hendrik (Schmick), although you were only around for the year of your master thesis, it was a lot of fun to supervise you! The same holds for Jannes (Schmitz), even though it was even shorter with his Bachelor thesis.

During my years of the thesis I spent a significant chunk of time at CERN. I want to extend my gratitude to everyone in the CAST collaboration. You're a great bunch of people! Thank you for being welcoming and supportive. Konstantin (Zioutas), Giovanni (Cantatore), Horst (Fischer), your focus was elsewhere in those last years of CAST of course, but I appreciated your feedback and it was fun working with you! Theodoros (Vafeiadis), a special thanks goes to you for not only being an excellent technical coordinator, but also just being fun and helpful outside of work! Marios (Maroudas) and Justin (Baier) thank you two as well. And Cristian (Cogollos) and Sergio (Arguedas) I loved our time together sharing "the corridor". You know who you are!

While I am not part of the group of Igor (García Irastorza) in Zaragoza, I spent a lot of time in the last few years in Zaragoza. Everyone of you treated me as if I *was* part of your group (aside from being IAXO collaboration members!). Thank you all! Special thanks go out to Julia (Vogel) and Jaime (Ruz)

for their efforts in helping me understand the LLNL telescope! And Konrad (Altenmüller), thanks for 'abseiling' into my life via your postdoc, haha! And Igor, thank you for helping me untangle the limit calculation.

On a completely different side of things, thanks to Andreas (Rumpf, @araq) for inventing the Nim programming language. Without you my thesis would certainly be different! Generally, the Nim community is a great bunch of really talented people (give Nim a try!). Too many to list here, but I want to highlight a few. Mamy (Ratsimbazafy, @mratsim) your work is extremely appreciated, not only by me as you very well know. And I'm especially grateful to you, for being as trusty as you were right from the start. You didn't know me, but you treated me with respect and collaborated with me, which gave me a huge boost in confidence to put my own code out there! Brent (Pedersen, @brentp), thanks for developing `nim-plotly`. It was really helpful in getting started to use Nim for my work! The same goes for Yuriy (Glukhov @yglukhov), being able to interact seamlessly with Python from Nim via `nimpy` was exceptionally useful. Regis (Caillaud, @clonkk) and Hugo (Granström @hugogranstrom), I love collaborating with you guys. Thanks for working with me on `SciNim`. Onto the future! And finally, Chuck (Charles Blake @c-blake), thank you for being a mentor and friend. :)

I consider most of the people I named above good friends, but mention you in a context that somehow relates to the thesis. A few people should be mentioned though that don't really fit in like that!

First, David (Helten), I hope we manage to stay in contact better again in the future! Stephan (Kürten), a big thank you to you, too (and of course for proofreading!). And Roberto (Röll), thanks for always being there for me (and reading the thesis!).

And finally, thanks to my family. Danke, Bianca, Papa und Mama, dass es euch gibt! And thank you Cristina (Margalejo) for being my partner and being at my side!