Numerical Methods for Uncertainty Quantification in Gas Network Simulation

Dissertation

zur

Erlangung des Doktorgrades (Dr. rer. nat.)

der

Mathematisch-Naturwissenschaftlichen Fakultät

der

Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt von

Barbara Fuchs

aus

Bad Honnef

Bonn 2018

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

1. Gutachter: Prof. Dr. Jochen Garcke

2. Gutachter: Prof. Dr. Marc Alexander Schweitzer

Tag der Promotion: 07. September 2018

Erscheinungsjahr: 2018

Zusammenfassung

Erdgas leistet einen wichtigen Beitrag zur deutschen Energieversorgung und wird hauptsächlich benötigt, um den Wärmebedarf von Industrie und Privathaushalten zu decken. Damit Gasnetze sicher und vor allem zuverlässig betrieben werden können, ist es wichtig verschiedenste Szenarien mit Hilfe von Simulationen durchzuspielen. Dabei treten diverse Unsicherheiten auf, zum einen in Modellparametern und zum anderen in Randbedingungen. Von großem Interesse ist der Einfluss des schwankenden Bedarfs an mehreren Entnahmestellen, da nicht klar ist, ob das Netzwerk alle Bedarfsspitzen decken kann. Bei dieser Form der Quantifizierung von Unsicherheiten geht es um die Vorwärtsanalyse, das heißt um die Frage, in wie weit unsichere Eingangsgrößen bestimmte Ausgangsgrößen beeinflussen.

Zur Beantwortung dieser Frage werden in dieser Arbeit zunächst die Gleichungen hergeleitet, welche den Gasfluss durch einzelne Rohre oder andere Elemente wie Absperrventile, Druckregelventile oder Kompressoren beschreiben. Durch die zusätzliche Massenerhaltung an Verbindungsstellen entsteht ein Gleichungssystem, welches den Gasfluss durch das komplette Gasnetz beschreibt.

Das zweite Kapitel beschäftigt sich mit mehreren Methoden zur Quantifizierung von Unsicherheiten. Möchte man nur verschiedene Statistiken der Lösung wie Erwartungswert oder Varianz bestimmen, so können herkömmliche Methoden zur numerischen Integration benutzt werden. Geht es darum, die komplette Lösung zu approximieren, so benötigt man entweder stochastische Galerkin-Verfahren oder stochastische Kollokationsverfahren. Stochastische Galerkin-Verfahren bezeichnet man als intrusiv, da eine schwache Formulierung des ursprünglichen Problems gelöst wird und deswegen existierender Code nicht genutzt werden kann. Dahingegen sind stochastische Kollokationsmethoden nicht-intrusiv. Sie approximieren eine unsichere Lösung mittels Interpolation durch mehrere Auswertepunkte. Die Lösung in diesen Punkten kann mit einem bestehenden Löser berechnet werden. Da wir einen Löser für Gasnetze haben und diesen auch benutzen möchten, interessieren wir uns also besonders für Kollokationsverfahren. Diese haben alle eine Gemeinsamkeit: Je glatter die Funktion ist, desto höhere Konvergenzraten können erreicht werden. Ist die Funktion weniger glatt, besitzt sie also Knicke oder Sprünge, dann ist die Konvergenzrate üblicherweise schlechter.

Da bei der Simulation des Gasflusses aber Knicke in der Lösung auftreten, stellen wir im nächsten Kapitel die stochastische Simplex-Kollokation vor. Dabei wird der Parameterraum mit Simplizes diskretisiert und die Lösung stückweise durch Poly-

nome approximiert. Da wir nach einem Simulationslauf wissen, ob ein knickverursachendes Druckregelventil aktiv war oder nicht, können wir die Funktion auf beiden Seiten des Knicks separat approximieren und erhalten dadurch eine explizite Approximation an den Knick. Durch diese Änderung des ursprünglichen Verfahrens ist es möglich die anfangs erhofften Konvergenzraten zu erreichen. Neben einem Beweis einer algebraischen Konvergenzrate wird sie zusätzlich an synthetischen Testfunktionen verifiziert. Außerdem stellen wir zwei neue Fehlerschätzer vor, welche für eine adaptive Verfeinerung der Triangulierung benötigt werden. Wir untersuchen die Verteilung des Fehlerschätzers auf den Simplizes und begründen damit, dass es sinnvoll ist, mehrere Simplizes auf einmal zu verfeinern.

Zum Schluss wenden wir das Verfahren der stochastischen Simplex-Kollokation auf ein reales Gasnetz an. Wir berechnen verschiedene Statistiken der Lösung und zeigen die dazugehörigen Konvergenzraten. Da die Lösung weniger glatt ist als erwartet, können die theoretischen Konvergenzraten nicht erreicht werden. Es stellt sich heraus, dass die Lösung neben den durch Druckregelventile verursachten Knicken auch noch Sprünge in den zweiten Ableitungen hat. Diese Sprünge haben keine physikalischen sondern numerische Gründe. Theoretisch könnten die Sprünge verhindert werden, aber das würde die globale Konvergenz des Lösers beeinflussen. Da dieser für die industrielle Anwendung hinreichend genau ist, besteht kein Grund ihn an dieser Stelle zu verändern. Da aber alle anderen Methoden auch unter diesen Sprüngen leiden, erreicht die stochastische Simplex-Kollokation dennoch die besten Ergebnisse und erreicht mit wenigen Punkten eine Genauigkeit in der Größe des Modellfehlers.

Contents

1 Introduction

Today, natural gas contributes significantly to Germany's energy supply and is mainly used to provide useful heat in industry and residential buildings. Natural gas is more climate-friendly than other fossil fuels, as its use is accompanied by lower CO_2 emissions. In addition, gas-fired power plants can be started up much faster than coal-fired power plants, making them ideal for compensating for electricity fluctuations from renewable energy sources. Germany produces only a low amount of natural gas itself and most of it is imported. Because of this, not only a distribution network is necessary but also long supply pipes. The German gas network consists of pipes with a total length of over 500,000 km. These pipes enable the safe delivery of largely variable quantities of gas over long distances. A large number of scenario analyses are necessary to ensure a secure and reliable operation of the network. Since not all of these scenarios can be tested, they are replaced by cheaper and faster simulations. Also in many other applications in engineering and science, numerical simulations are used to replace expensive and time consuming physical experiments.

"As far as the propositions of mathematics refer to reality, they are not certain; and as far as they are certain, they do not refer to reality."

Geometry and Experience, Lecture before the Prussian Academy of Sciences, January 27, 1921

Albert Einstein

Mathematical simulations typically involve several types of errors and uncertainties. On the one hand, errors in the model often arise by simplifying the exact physics to reduce the complexity of the simulation. For example, in gas flow simulations it is common to model the pressure loss due to friction along the pipe with the Darcy-Weisbach equation [KHPS15, Lur08, SSW16]. This equation is only valid for incompressible fluids, but, of course, gas is compressible. On the other hand, uncertainties in the input data may concern the system's geometry, the boundary and initial conditions, or the model coefficients. Often, the data cannot be exactly determined, e.g. the roughness at each point of a pipe or the temperature of the soil. These values cannot be measured everywhere and therefore induce an epistemic uncertainty. In this thesis, we will not consider these types of inaccuracies,

as there are common assumptions and models for them. We are more interested in aleatory uncertainties when the value of a variable differs each time we run the same experiment. The best example here are the costumers that withdraw different amounts of gas at different times. It is of great interest whether the gas network can meet the demand when all customers need a lot of gas at once and how likely a failure is. Therefore, uncertainty quantification is often about how uncertainties in the input data influence certain output variables, the quantities of interest. For this forward propagation we need methods to approximate and integrate high-dimensional functions.

The Uncertainty Quantification

Nearly all methods incorporate the finite noise assumption, i.e. one assumes that all uncertainty can be represented by a finite number of independent random variables. The Karhunen-Loève expansion [ES14, AGP$^+$08, CGST11, BD17, GKW$^+$07] of a random field provides a series representation in terms of its spatial correlation. The resulting uncorrelated coefficients can further be expressed as functions of independent random variables. Therefore, the truncated Karhunen-Loève expansion is the standard preprocessing method to obtain a finite noise. If finite noise is ensured, we have two types of uncertainty quantification: stochastic Galerkin methods and stochastic collocation methods.

In a stochastic Galerkin method, the polynomial chaos expansion [PNI15, MK10] of the solution is usually calculated first. The polynomial chaos expansion is a spectral expansion in terms of orthogonal polynomials with respect to the density distribution of the uncertain variables and it decouples random and spatial dimensions. Next, one takes a Galerkin projection [MK10, TMEP11, BTNT12] of the original problem equation onto each of the orthogonal basis polynomials which yields a weak formulation of the original problem. This intrusive method yields an exponential convergence for sufficiently smooth solutions but has the disadvantage that the original deterministic system must be modified into a larger system of coupled equations. Moreover, a probably existing solver for the original problem cannot be reused.

If already an efficient solver for the deterministic problem exists, non-intrusive stochastic collocation methods are the methods of choice because they only incorporate solutions of the original problem. An uncertain solution is approximated by interpolating several sampling points. Typically a Lagrange interpolation is used, but piecewise polynomial interpolation is also possible. Stochastic collocation methods have not only the advantage that the original solver can be reused, but, moreover, the samples are independent of each other and can therefore be calculated in parallel. If the solution is sufficiently smooth, also stochastic collocation methods such as sparse grid interpolation [BTNT12, ES14, FP16] can achieve a fast convergence. For the computation of statistics of the solution, such as expectation and variance of the solution, standard methods for numerical integration can be used. In the case of smooth functions the range of methods is wide: Gaussian quadrature [KW16, TI14, AV13], sparse grid quadrature [BTNT12, ES14, FP16], or (quasi-) Monte Carlo [SST17, CGST11, CGP17] quadrature can be used. The smoother the

integrand is, the higher are the convergence rates that can be achieved. The variety of methods is much smaller if the function is not smooth. Monte Carlo integration can, of course, always be used. Additional methods for discontinuous functions are spatially adaptive sparse grids [JAX11, Pfl10, Pfl12, GK14], Voronoi piecewise surrogate models [RSP$^+$17], or simplex stochastic collocation [WI12a, WI12b, WI13]. The ideas behind Voronoi piecewise surrogate models and simplex stochastic collocation are quite similar. In both cases the function is locally approximated by piecewise polynomials either on Voronoi cells or on simplices resulting from a Delaunay triangulation. In the Voronoi piecewise surrogate model a jump in the function is detected if the difference in the function values between neighboring cells exceeds a user defined threshold, whereas in simplex stochastic collocation a jump is not directly recognized but the resulting oscillations in the interpolation. Non-smooth functions with kinks can be smoothed by integration [GKS13, GKS17] over one dimension if the location of the kink is known a-priori. Unfortunately, it is not possible to predict the locations of kinks arising in gas network simulations.

The Gas Network

A gas network is modeled with nodes and edges. The edges represent pipes or other network elements such as valves, control valves, heaters, or compressors. Gas flow through a single pipe is described by the Euler equations, a set of partial differential equations [KHPS15, Lur08, SSW16]. The first equation is the continuity equation following from the conservation of mass, whereas the law of momentum conservation specifies the pressure loss along the pipe due to weight, pressure, and frictional forces. The equation of state is necessary to describe the state of a real compressible gas for a given set of values for temperature, density, and pressure. The first law of thermodynamics must be taken into account to describe any heat transfer process. A solution to this system of equations can be found analytically if we assume a stationary and isothermal gas flow [SSW16]. Analogously to Kirchhoff's law, the mass must be conserved at junctions where several pipes are connected, whereas at supply nodes the incoming gas pressure is given and at demand nodes the extracted mass flow. If a gas network consists of pipes only, the solution of the pressure, density, and temperature at nodes and the gas flow in pipes is sufficiently smooth. But a real gas network also contains even more complicated elements. Usually, the pressure in transport pipes is significantly larger than the maximum allowable operating pressure in distributional pipes. Due to this reason the network needs pressure control valves that adjust the outgoing pressure if the incoming pressure exceeds a preset limit. Unfortunately, the more complicated elements impair the smoothness of the solution. For example, a pressure control valve causes kinks, i.e. locations where the function is not differentiable, in the solution. Increasing the pressure at a supply node increases the pressure after a control valve until the preset pressure is reached, but afterwards the pressure remains constant, see Figure 1.1. We do not know in advance where the kink is located, but after the simulation run we know if a control valve has regulated the pressure or not. Using this information we are able to improve the convergence rate of the original simplex stochastic collocation.

The idea of simplex stochastic collocation is to approximate a function $f : [0, 1]^d \to \mathbb{R}$

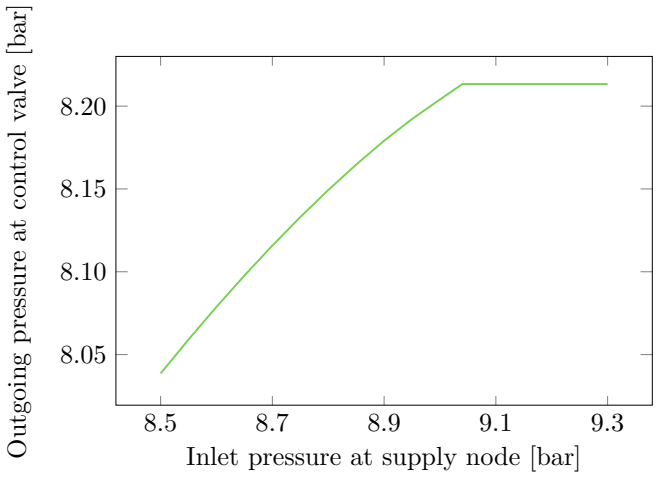


Figure 1.1: A kink in the solution resulting from pressure regulation.

by a piecewise polynomial interpolation on simplices. In case of discontinuities polynomial interpolation becomes oscillatory, this phenomenon is also known as the Gibbs phenomenon. To avoid oscillations one ensures that the approximation is local extremum conserving, i.e. maximum and minimum of the approximation in any simplex must be attained at its vertices, otherwise the polynomial degree p is decreased by one [WI12a, WI12b, WI13]. This condition results in a fine discretization near discontinuities and a coarser discretization at smooth regions. We investigated the original approach for functions with kinks but were not able to reach the theoretical [SX95] algebraic convergence rates of $\mathcal{O}(-\frac{p+1}{d})$ because the approximation of a kink does not improve by using higher degree polynomials. Instead, a kink can be better approximated by incorporating the information if a control valve is active or not. With this information we can approximate the solution on both sides of the kink separately. The kink itself is approximated by taking the minimum or maximum, respectively, of both functions. Since the function is sufficiently smooth on both sides, we obtain a better convergence with the theoretical algebraic convergence rates.

Outline of this Thesis

The remainder of this thesis is organized as follows:

- The second chapter deals with the simulation of the gas flow through a network of pipes and other elements. We derive the equations describing the gas flow in a single pipe element and show how to model other network elements, such as valves, control valves, compressors, or heaters. These equations together with the mass conservation at junctions form a system of equations which describes the gas flow through a complete network.

- Chapter 3 is concerned with uncertainty quantification. After a short classification of typical problems, we give a brief overview on some aspects of probability theory which are essential for a correct problem description in uncertainty quantification. Then we discuss several methods for numerical

integration, such as Gauss, sparse grid, Monte Carlo, and quasi-Monte Carlo quadrature. These methods are not special methods for uncertainty quantification, but in this context they are used for the computation of expected values and variances. In contrast, the next sections are concerned with the two most frequently used methods in the field of uncertainty quantification. On the one hand, there are intrusive Galerkin methods that provide a spectral convergence but require a modification of the original problem, and on the other hand, there are non-intrusive stochastic collocation methods which only incorporate several solutions of the original problem. Both methods, along with important features, are described and compared.

- In Chapter 4, we introduce the method of stochastic simplex collocation for uncertainty quantification in gas network simulation. The use of this method is motivated by the kinks in the solution, e.g. due to pressure control valves. We analyze the original version which is intended for functions with jumps and modify it so that it becomes applicable to functions with kinks. For this new modified version we prove an algebraic convergence rate and verify it with by a synthetic function. We derive two new error estimators for an adaptive refinement and compare them with an already existing error estimator. Moreover, we study the distribution of the error estimator over the simplices and show that multiple refinements are possible and reasonable. Lastly, we compare the new stochastic simplex collocation method to the similar Voronoi piecewise surrogate models.

- Chapter 5 includes numerical results for the application of simplex stochastic collocation to the solution of gas network simulation. We compute several statistics of our quantity of interest and show the corresponding convergence plots. Arising problems due to the used gas network solver are analyzed and discussed in detail. The results of stochastic simplex collocation are compared to standard methods like Monte-Carlo and quasi-Monte Carlo methods.

- Finally, we conclude this thesis in Chapter 7 by providing a summary and an outlook regarding possible extensions of the stochastic simplex collocation method for uncertainty quantification in gas network simulation.

2 Simulation of Gas Networks

Under the surface hundreds kilometers of pipelines can be found supplying gas to industrial and private consumers. Gas is transported through a network from one point to another. Providers pump gas at supply nodes into the network, which is then drawn at demand nodes by costumers. It is fixed by contract under which pressure gas is fed in to the network or which mass flow exists at demand nodes. To achieve these agreements we require a network control with valves, compressors, and regulators. In addition, the control should be as cost-efficient as possible. This problem can be solved by numerical optimization, but first we need to model the gas flow in a network.

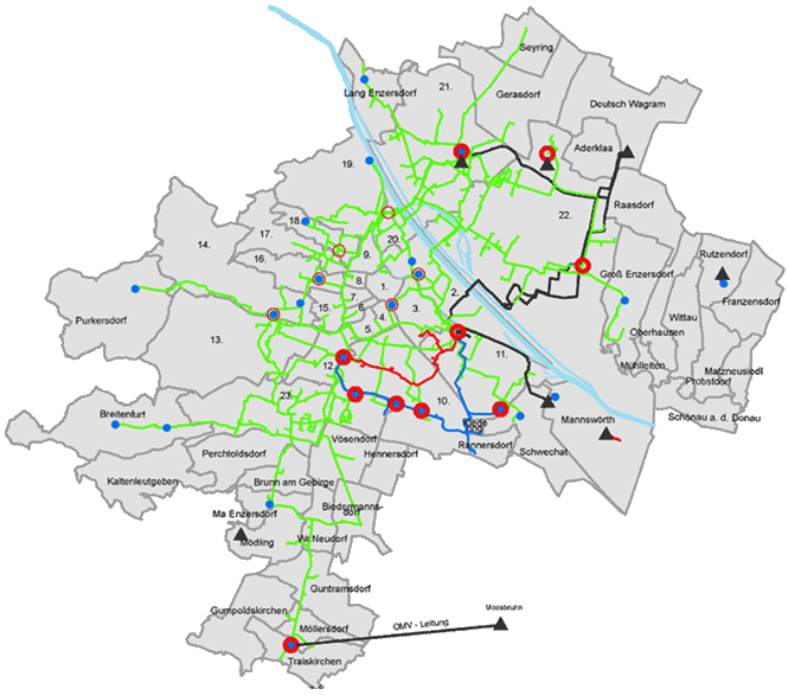


Figure 2.1: An example of a supply network[1].

[1]Image source: https://de.wikipedia.org/wiki/Datei:Versorgungsgebiet_Gasnetz.png

2.1 Euler Equations for Gas Flow in Pipes

Gas transport within a single pipe segment is modeled with the so called Euler equations. To characterize a pipe's geometry we need parameters like length L, diameter D, geodesic height h, as well as its roughness k at the pipe's inner side. Let $t \geq 0$ be the time and $x \in [0, L]$ the position at a pipe segment. Then, at time t and position x, we have the gas density $\rho(x,t)$, pressure $p(x,t)$, temperature $T(x,t)$, and flow velocity $v(x,t)$. The mass flow $q(x,t) = \rho(x,t)v(x,t)A$ describes the amount of gas mass per time flowing through the pipe with cross-sectional area A. In the following, for simplicity's sake, we neglect the dependence of the considered quantities on place x and time t.

symbol	**name**	**unit**
$h(x,t)$	geodesic height	m
D	pipe diameter	m
A	cross-sectional area	m^2
L	pipe length	m
k	pipe roughness	mm
$\rho(x,t)$	gas density	$\mathrm{kg/m}^3$
$p(x,t)$	gas pressure	$\mathrm{kg/ms}^2$
$T(x,t)$	gas temperature	K
$v(x,t)$	gas velocity	m/s
$q(x,t)$	mass flow	kg/s

Table 2.1: Physical quantities of the pipe and the gas.

2.1.1 The Continuity Equation

The law of conservation of mass states that mass can neither be created nor destroyed. Thus a system's mass stays constant. For a gas flowing through a pipe, the difference between the mass flow into and out of a control volume equals the mass change within the control volume, cf. [DW76]. See Figure 2.2 for an illustration of the control volume. At the fixed time t, the mass flow into the control volume through the cross-sectional area A_1 at x_1 equals

$$q_{\text{in}} = \rho(x_1,t)v(x_1,t)A.$$

Using the fundamental theorem of calculus, the mass flow out of the control volume through the cross-sectional area A_2 at x_2 can be written as

$$q_{\text{out}} = \rho(x_1,t)v(x_1,t)A + \int_{x_1}^{x_2} \partial_x(\rho v)A \,\mathrm{d}x.$$

Hence, on the one hand, the difference between q_{out} and q_{in} is given by

$$q_{\text{out}} - q_{\text{in}} = \int_{x_1}^{x_2} \partial_x(\rho v)A \,\mathrm{d}x.$$

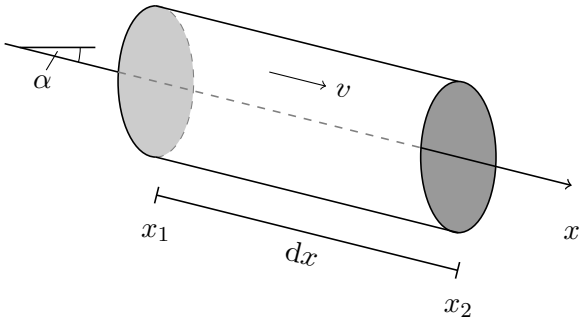


Figure 2.2: The one-dimensional flow through a control volume.

On the other hand, integrating the product of density and area over x yields the mass within the control volume

$$m = \int_{x_1}^{x_2} \rho(x,t) A \, \mathrm{d}x,$$

which changes with a rate of $\partial_t \int \rho A \, \mathrm{d}x$. By interchanging differentiation and integration we obtain

$$\partial_t \, m = \partial_t \int_{x_1}^{x_2} \rho(x,t) A \, \mathrm{d}x = \int_{x_1}^{x_2} \partial_t \rho A \, \mathrm{d}x.$$

This rate of change must be equal to $q_{\text{out}} - q_{\text{in}}$ and therefore

$$\int_{x_1}^{x_2} \partial_x (\rho v) A \, \mathrm{d}x = \int_{x_1}^{x_2} \partial_t \rho A \, \mathrm{d}x.$$

Because this equation must hold true for any volume, we can omit the integral and dividing by A yields

$$\partial_t \rho + \partial_x (\rho v) = 0. \tag{2.1}$$

This equation is called *continuity equation*.

2.1.2 The Law of Conservation of Momentum

Following Newton's second law, the change over time of a body's momentum equals the sum of the external forces acting on this body

$$\sum F_x = \frac{\mathrm{d}}{\mathrm{d}t}(mv).$$

In our case, the considered body is a gas in some control volume between $x_1(t)$ and $x_2(t)$ flowing in x-direction through the pipe, see Figure (2.2). We choose the control volume so small that we can assume uniform velocity. Weight force, pressure force, and frictional force are acting on the gas, cf. [DW76].

The Weight Force

The part of the weight force acting parallel to the inclined plane is calculated from the mass m, the gravitational constant g, and pipe's inclination α below the horizontal

$$\begin{aligned} F_A &= -gm\sin(\alpha) \\ &= -\int_{x_1(t)}^{x_2(t)} g\rho(x,t)A\sin(\alpha)\,\mathrm{d}x \\ &= -\int_{x_1(t)}^{x_2(t)} g\rho(x,t)\partial_x h(x,t)A\,\mathrm{d}x. \end{aligned}$$

The Pressure Force

The pressure force is given by

$$\begin{aligned} F_D &= p(x_1)A - p(x_2)A \\ &= -\int_{x_1(t)}^{x_2(t)} \partial_x p(x)A\,\mathrm{d}x. \end{aligned}$$

The Frictional Force

The friction within the pipe causes a pressure loss from the beginning to the end of the control volume. Following the Darcy-Weisbach equation [Bro02] the pressure loss $p(x_1) - p(x_2)$ is given by

$$p(x_1) - p(x_2) = \lambda\frac{\mathrm{d}x}{2D}\rho|v|v, \tag{2.2}$$

with the friction factor λ and the mean velocity v. This equation only holds true for incompressible fluids, but in our application of compressible flow where the calculated pressure loss is less than 10% of the absolute pressure $|p(x_1)|$, it is sufficiently accurate. For more details on the restrictions see [Cra09]. Thus the frictional force is given by

$$\begin{aligned} F_R &= -\int_{x_1(t)}^{x_2(t)} \partial_x p(x)A\,\mathrm{d}x \\ &= -\frac{\lambda}{2D}\rho|v|vA\,\mathrm{d}x. \end{aligned}$$

The friction factor λ is a dimensionless quantity which is needed for determining the pressure loss due to friction at the inside of the pipe. In case of hydraulically rough pipes where the irregularities are not covered by a viscous sublayer, λ can be either calculated with the formula by Nikuradse [Nik33]

$$\lambda = \left(1.138 + 2\log\left(\frac{D}{k}\right)\right)^{-2}$$

or the more precise formula by Hofer [Hof73]

$$\lambda = \left(-2\log\left(\frac{4.518}{Re(q)}\log\left(\frac{Re(q)}{7}\right) + \frac{k}{3.71D}\right)\right)^{-2}, \tag{2.3}$$

where k denotes the pipe's roughness, D its diameter, and $Re(q)$ the Reynolds number. The Reynolds number is defined as

$$Re(q) = \frac{D}{A\eta}|q|,$$

where η denotes the dynamic viscosity of the gas. The flow is called turbulent if $Re(q) \gtrsim 2320$ and laminar otherwise. For high Reynolds numbers $Re(q) \to \infty$, the Hofer equation approaches the easier Nikuradse equation.

The Change of Momentum

Now we calculate the last missing term, the system's change of momentum $\frac{\mathrm{d}}{\mathrm{d}t}mv$. As above, we derive the mass of the gas by integrating over the control volume

$$mv = \int_{x_1(t)}^{x_2(t)} \rho(x,t)Av \,\mathrm{d}x.$$

Note that we assumed uniform velocity. To differentiate this parameter integral, we need the following Newton-Leibniz theorem:

Theorem 2.1 (Newton-Leibniz)**.** *Let $f : [x_1, x_2]\times]t_1, t_2[\to \mathbb{R}$ be a continuous function and for each fixed $x \in [x_1, x_2]$ let the mapping $t \mapsto f(x,t)$ be differentiable by t $\forall t \in]t_1, t_2[$ and let the partial derivative $\partial_t f : [x_1, x_2]\times]t_1, t_2[\to \mathbb{R}$ be also continuous. Then*

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{x_1(t)}^{x_2(t)} f(x,t)\,\mathrm{d}x = \int_{x_1(t)}^{x_2(t)} \partial_t f(x,t)\,\mathrm{d}x + \int_{x_1(t)}^{x_2(t)} \partial_x f(x,t)v(x,t)\,\mathrm{d}x.$$

Proof. Define $\psi(u,v,t) := \int_u^v f(x,t)\,\mathrm{d}x$. Using the dominated convergence theorem we can interchange integration and differentiation, thus

$$\partial_t\psi(u,v,t) = \int_u^v \partial_t f(x,t)\,\mathrm{d}x.$$

We can write the integral in terms of ψ

$$\int_{x_1(t)}^{x_2(t)} f(x,t)\,\mathrm{d}x = \psi(x_1(t), x_2(t), t)$$

and obtain by the chain rule

$$\frac{\mathrm{d}}{\mathrm{d}t}\psi(x_1(t), x_2(t), t) = \partial_{x_1}\psi \cdot \frac{\mathrm{d}}{\mathrm{d}t}x_1(t) + \partial_{x_2}\psi \cdot \frac{\mathrm{d}}{\mathrm{d}t}x_2(t) + \partial_t\psi.$$

From the fundamental theorem of calculus it follows that $\partial_v \psi = f(v,t)$ and $\partial_u \psi = -f(u,t)$. In addition we use the fact that $\mathrm{d}x/\mathrm{d}t = v$. Hence,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_{x_1(t)}^{x_2(t)} f(x,t)\,\mathrm{d}x &= \partial_{x_1}\psi(x_1,x_2,t)\cdot v_1(t) + \partial_{x_2}\psi(x_1,x_2,t)\cdot v_2(t) + \partial_t\psi \\ &= -f(x_1,t)v_1(t) + f(x_2,t)v_2(t) + \partial_t\psi \\ &= \int_{x_1(t)}^{x_2(t)} \partial_x f(x,t)v(t)\,\mathrm{d}x + \int_{x_1(t)}^{x_2(t)} \partial_t f(x,t)\,\mathrm{d}x.\end{aligned}$$

□

With this theorem we obtain

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}mv &= \frac{\mathrm{d}}{\mathrm{d}t}\int_{x_1(t)}^{x_2(t)} \rho(x,t)Av\,\mathrm{d}x \\ &= \int_{x_1(t)}^{x_2(t)} \partial_x\rho(x,t)Av^2\,\mathrm{d}x + \int_{x_1(t)}^{x_2(t)} \partial_t\rho(x,t)Av\,\mathrm{d}x \\ &= \int_{x_1(t)}^{x_2(t)} \partial_x\rho(x,t)Av^2\,\mathrm{d}x + \int_{x_1(t)}^{x_2(t)} \partial_t q(x,t)\,\mathrm{d}x\end{aligned}$$

Incorporating all terms, we obtain the law of conservation of momentum

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}(mv) &= \sum F_x \\ \int_{x_1(t)}^{x_2(t)} \partial_t q\,\mathrm{d}x + \int_{x_1(t)}^{x_2(t)} \partial_x(\rho A v^2)\,\mathrm{d}x &= -\int_{x_1(t)}^{x_2(t)} g\rho\partial_x h A\,\mathrm{d}x - \int_{x_1(t)}^{x_2(t)} \partial_x pA\,\mathrm{d}x \\ &\quad - \int_{x_1(t)}^{x_2(t)} \frac{\lambda}{2D}\rho|v|vA\,\mathrm{d}x,\end{aligned}$$

and again after omitting the integral and dividing by A

$$\frac{1}{A}\partial_t q + \partial_x(\rho v^2) + g\rho\partial_x h + \partial_x p + \frac{\lambda}{2D}\rho|v|v = 0. \tag{2.4}$$

2.1.3 The Equation of State for Real Gases

Adding the compressibility factor z to the thermodynamic equation of state for an ideal gas, we are capable of describing the behavior of real gases by

$$p = z(p,T)\rho R_s T \tag{2.5}$$

with the specific gas constant R_s. Several formulas approximate the compressibility factor differing in accuracy, complexity, and validity range for temperature and pressure.

The AGA Formula

An easy way of approximating the compressibility of a gas with a given pressure p and a temperature T provides the formula by the American Gas Association (Report 8)

$$z(p,T) = 1 + \alpha(T)p \tag{2.6}$$

with

$$\alpha(T) = \alpha_\infty - \frac{\beta}{T},$$

$\alpha_\infty = 0.257/p_c$, and $\beta = 0.533T_c/p_c$. The critical temperature T_c and the critical pressure p_c of the gas characterize the critical point above which the physical states liquid and gaseous cannot be distinguished. The AGA formula is linear in pressure and provides reliable results up to a pressure of 70 bar.

Papay's Formula

A more precise approximation provides Papay's formula [Pap68] by

$$z(p,T) = 1 - \alpha(T)p + \beta(T)p^2 \tag{2.7}$$

with

$$\alpha(T) = \alpha_p 10^{-\alpha_T T}$$
$$\beta(T) = \beta_p 10^{-\beta_T T},$$

and $\alpha_p = 3.52/p_c$, $\alpha_T = 0.9813/T_c$, $\beta_p = 0.274/p_c^2$, and $\beta_T = 0.8157/T_c$. This formula incorporates not only a linear term in p, but also a quadratic one and can be used up to a pressure of 150 bar. See Figure 2.3 for the differences between both formulas.

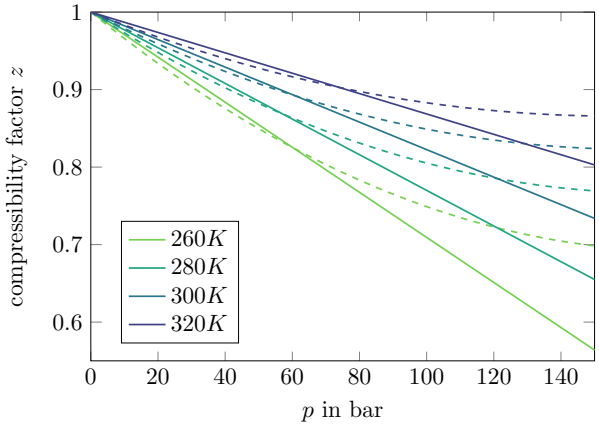


Figure 2.3: The compressibility factor z versus pressure in bar for methane computed with AGA Formula (solid) and Papay's Formula (dashed). Critical temperature: 190.6 Kelvin, critical pressure: 46 bar.

2.1.4 The Conservation of Energy

The first law of thermodynamics states that the total energy of a system is conserved. Energy does not appear or disappear from anywhere, but it changes from one form into another. Following [Lur08] the change in the total energy equals the sum of the external inflow of heat and the work of the external forces

$$\frac{\mathrm{d}(E_{\mathrm{kin}} + E_{\mathrm{in}})}{\mathrm{d}t} = \frac{\mathrm{d}Q_{\mathrm{ex}}}{\mathrm{d}t} + \frac{\mathrm{d}F_{\mathrm{ex}}}{\mathrm{d}t}. \tag{2.8}$$

Consider a movable control volume of transported gas enclosed between two cross-sections $x_1(t)$ and $x_2(t)$ and let $e_{\mathrm{in}} = E_{\mathrm{in}}/m$ denote the internal energy of a unit mass of the volume, then the first term of equation (2.8) can be written as

$$\begin{aligned}\frac{\mathrm{d}(E_{\mathrm{kin}} + E_{\mathrm{in}})}{\mathrm{d}t} &= \frac{\mathrm{d}}{\mathrm{d}t}\left[\int_{x_1(t)}^{x_2(t)} \left(\frac{\rho v^2}{2} + \rho e_{\mathrm{in}}\right) A \,\mathrm{d}x\right] \\ &= \int_{x_1(t)}^{x_2(t)} \partial_t\left[\left(\frac{v^2}{2} + e_{\mathrm{in}}\right)\rho A\right] + \partial_x\left[\left(\frac{v^2}{2} + e_{\mathrm{in}}\right)\rho A v\right] \mathrm{d}x\end{aligned}$$

where we used the Newton-Leibniz Theorem 2.1.2 to differentiate the integral. The second and third term of equation (2.8) can be written as

$$\begin{aligned}\frac{\mathrm{d}Q_{\mathrm{ex}}}{\mathrm{d}t} &= \int_{x_1(t)}^{x_2(t)} \pi D q_n \,\mathrm{d}x \\ \text{and} \quad \frac{\mathrm{d}F_{\mathrm{ex}}}{\mathrm{d}t} &= -\int_{x_1(t)}^{x_2(t)} \partial_x(pAv)\,\mathrm{d}x - \int_{x_1(t)}^{x_2(t)} \rho g \partial_x h v A \,\mathrm{d}x\end{aligned}$$

where q_n is the heat flux going through the unit area of the pipeline surface per unit time. This heat flux is usually modeled with the Newton formula

$$q_n = -c_{\mathrm{HT}}(T - T_{\mathrm{ex}})$$

by which the flow is proportional to the difference between the temperature T of the gas and the temperature T_{ex} outside the pipe. The factor c_{HT} is called heat-transfer factor. Since equation (2.8) must hold true for any control volume, the integral can be omitted and we obtain the differential equation

$$\partial_t\left[\left(\frac{v^2}{2} + e_{\mathrm{in}}\right)\rho A\right] + \partial_x\left[\left(\frac{v^2}{2} + e_{\mathrm{in}} + \frac{p}{\rho}\right)\rho A v\right] + \pi D c_{\mathrm{HT}}(T - T_{\mathrm{ex}}) + \rho g \partial_x h v A = 0.$$

The inner energy of the gas can be calculated with

$$e_{\mathrm{in}} = c_v T + \mathrm{const}$$

where c_v denotes the specific heat capacity at constant volume.

2.2 Gas Network Elements

2.2.1 Pipes

The modeling of the gas flow through a single pipe was already described in the previous section. But a real gas network does not only include pipes but also additional elements such as valves, resistors, or heaters, which are described in the following subsections. For a more detailed description of the following and further gas network elements see [KHPS15].

2.2.2 Valves

A valve is a switch which is either open or closed. If it is open, the gas flows through the valve and neither pressure nor mass flow are affected, i.e.

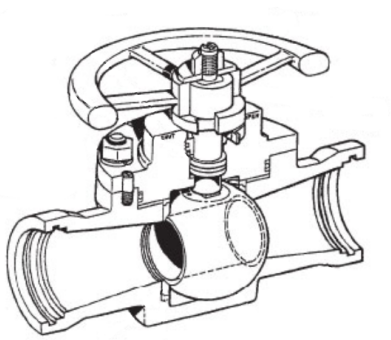
$$q_{\text{in}} = q_{\text{out}}$$
$$p_{\text{in}} = p_{\text{out}}.$$

If the valve is closed, no gas flow is possible, i.e.

$$q_{\text{in}} = q_{\text{out}} = 0.$$

In this case the pressures p_{in} and p_{out} on both sides of the valve are decoupled. See Figure 2.4 for a photograph and a schematic diagram of a ball valve, which is often used in gas networks. A ball valve is a form of quarter-turn valve which uses a hollow, perforated and pivoting ball to control the flow through it. It is open when the ball's hole is in line with the flow and closed when it is pivoted 90-degrees by the valve handle.

(a) Photograph.



(b) Schematic diagram.

Figure 2.4: A ball valve[2].

[2]Image sources: https://stock.adobe.com, http://informefebrerotanquehidraulico828826.blogspot.de/2015/04/valvulas-de-bloqueo-y-valvulas-de.html

2.2.3 Control Valves

Typically, different parts of a gas network are operated at different pressures. In larger transport pipes the pressure is higher than in distributional pipes with a smaller diameter. In order to connect these different pipes we need elements that can reduce the pressure, so-called control valves or pressure regulators. See Figure 2.5 for a photograph of a control valve. The degree of opening of the valve and hence the rate of flow is controlled by a diaphragm actuator in combination with a compression spring. The higher the outgoing pressure is, the more closed is the valve. Therefore, the outgoing pressure is regulated to a preset pressure p_{set}. Due to technical limitations, a control valve can only work in certain range, i.e. if

$$p_{\text{in}} \geq p_{\min}, \quad p_{\text{in}} \geq p_{\text{out}}, \quad p_{\text{out}} \leq p_{\text{set}} \leq p_{\max}, \quad \text{and} \quad 0 \leq q \leq q_{\max}.$$

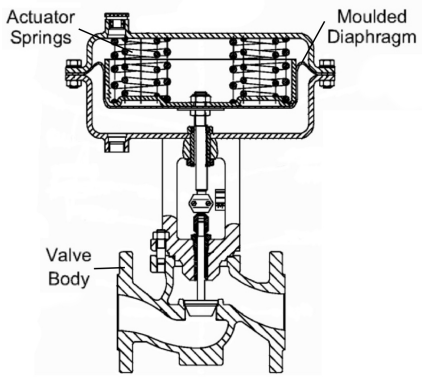
The Closed Mode

If the incoming pressure falls below $p_{\min}$, or the outgoing pressure exceeds p_{set} or the incoming pressure p_{in}, or the flow exceeds $q_{\max}$ or changes direction, then the control valve closes automatically:

$$\begin{aligned}
p_{\text{in}} < p_{\min} \quad &\Rightarrow \quad q = 0, \quad p_{\text{out}} \text{ arbitrary} \\
p_{\text{out}} > p_{\text{set}} \quad &\Rightarrow \quad q = 0, \quad p_{\text{in}} \text{ arbitrary} \\
p_{\text{out}} > p_{\text{in}} \quad &\Rightarrow \quad q = 0 \\
q > q_{\max} \quad &\Rightarrow \quad q = 0, \quad p_{\text{in}}, p_{\text{out}} \text{ arbitrary} \\
q < 0 \quad &\Rightarrow \quad q = 0, \quad p_{\text{in}}, p_{\text{out}} \text{ arbitrary.}
\end{aligned}$$

In this case the pressures p_{in} and p_{out} on both sides of the valve are decoupled.

(a) Photograph.



(b) Schematic diagram.

Figure 2.5: A pressure control valve[3].

[3]Image sources: https://stock.adobe.com, https://www.springer.com/de/book/9783540694700, adapted by permission from Springer Customer Service Centre GmbH: Springer Nature, Control of Actuators for Process Valves by Peter Beater, 2007 [Bea07]

The Bypass Mode

In the second operation mode – the bypass mode – the pressure of the gas is not affected. This is the case when all working bounds are met and the incoming pressure is already smaller or equal to p_{set}:

$$p_{\text{in}} \geq p_{\min}, \quad p_{\text{out}} \leq p_{\text{in}} \leq p_{\text{set}}, \quad 0 \leq q \leq q_h \qquad \Rightarrow \qquad p_{\text{out}} = p_{\text{in}}.$$

The Active Mode

If the incoming pressure, the outgoing pressure, and the mass flow lie in the working range of the valve, but the incoming pressure is higher than p_{set}, then the valve is active and controls the output pressure to p_{set}:

$$p_{\text{in}} \geq p_{\min}, \quad p_{\text{in}} > p_{\text{set}}, \quad p_{\text{out}} \leq p_{\text{set}}, \quad 0 \leq q \leq q_h \qquad \Rightarrow \qquad p_{\text{out}} = p_{\text{set}}.$$

This regulation will cause kinks in the resulting flows and pressures.

Analogously, it is possible to control the incoming pressure or the mass flow, but these types of regulators are not considered in this thesis.

2.2.4 Resistors

In addition to pressure loss caused by friction there are several more complicated network components like measurement devices, filter systems, curved pipes, or reduced radii that also induce a pressure loss which must be taken into account. These losses are modeled with a resistor as a surrogate. The pressure loss in a resistor is described according to the Darcy-Weisbach formula (2.2)

$$p_{\text{in}} - p_{\text{out}} = \frac{1}{2}\zeta\rho_{\text{in}}v|v|$$

where ζ is some pressure loss coefficient. The parameter ζ must be fitted to measurements of the pressure loss.

2.2.5 Heaters

At demand nodes the high pressure of a gas is reduced which causes an undesired cooling due to the Joule-Thomson effect. Thus a pre-heating of the gas is necessary to remain above the dew point after the pressure is reduced. The Joule-Thomson effect results from interactions between gas particles. If the pressure is reduced, the distance between the particles increases. Since the particles attract each other (Van-der-Waals forces), mechanical work must be done to overcome the attraction forces during expansion. Thereby the particles slow down and the gas cools down. This effect is described by the Joule-Thomson coefficient [SSW16]

$$\frac{\mathrm{d}T}{\mathrm{d}p} = \frac{RT^2}{\tilde{c}_p p}\frac{\partial z(p,T)}{\partial T}$$

where R denotes the universal gas constant and $\tilde{c}_p$ the molar heat capacity at constant pressure. This equation is approximated with a step of an implicit Euler method, which is sufficiently accurate

$$\frac{T_{\text{out}} - T_{\text{in}}}{p_{\text{out}} - p_{\text{in}}} = \frac{R}{\tilde{c}_p} \frac{T_{\text{out}}^2}{p_{\text{out}}} \frac{\partial z(p_{\text{out}}, T_{\text{out}})}{\partial T}$$

with temperatures $T_{\text{in}}, T_{\text{out}}$, and pressures $p_{\text{in}}, p_{\text{out}}$ at inlet and outlet pressures, respectively. The mass flow is not affected by the heater, i.e. $q = q_{\text{in}} = q_{\text{out}}$.

2.3 Isothermal, Stationary Networks

In the following we only consider the stationary case where the gas is in a steady state. In this case the gas flow is time-independent, i.e. all derivatives $\partial_t \cdot$ in the Euler equations are equal to zero. Thus the continuity equation (2.1) states that the mass flow along the pipe is constant, i.e.

$$\partial_x q = 0.$$

Then the momentum equation (2.4) reduces to

$$\partial_x p + g\rho\partial_x h + \frac{\lambda}{2D}\rho|v|v = 0. \tag{2.9}$$

Note that we omitted the term $\partial_x(\rho v^2)$ because under normal operating conditions it is very small in relation to the remaining terms and thus can be neglected. In addition, we assume that the gas temperature T and the compressibility factor z are constant along the pipe and can be approximated by some mean values T_m and z_m, respectively.

2.3.1 Explicit solution for $p(x)$

Following [KHPS15] we can solve the resulting linear ordinary differential equation (ODE) analytically by variations of constants.

Theorem 2.2. *For a constant slope $\partial_x h = s \neq 0$ of the pipe the solution $p(x)$ to (2.9) with the initial value $p(0) = p_{in}$ is given by*

$$p(x)^2 = \left(p_{in}^2 - \Lambda|q|q\,\frac{e^{Sx} - 1}{S}\right) e^{-Sx} \tag{2.10}$$

with

$$S := \frac{2gs}{R_s z_m T_m}, \quad \Lambda := \lambda \frac{R_s z_m T_m}{A^2 D}.$$

Proof. In (2.9) we replace the gas velocity v by the relation $\frac{q}{\rho A}$ and each appearing ρ using the equation of state of the gas (2.5). Hence we obtain

$$\partial_x p + g\frac{p}{R_s z_m T_m}s + \lambda\frac{|q|q}{2A^2D}\frac{R_s z_m T_m}{p} = 0$$

where we assumed a constant gas temperature T_m and a constant compressibility factor z_m. Multiplication by $2p$ results in

$$\partial_x p^2 + Sp^2 = -\Lambda|q|q.$$

Substituting $y = p^2$ yields a first-order linear ODE

$$\partial_x y + Sy = -\Lambda|q|q, \quad y(0) = p_{\text{in}}^2. \tag{2.11}$$

This ODE can be solved analytically by variation of constants and we obtain the solution

$$y(x) = p(x)^2 = \left(-\Lambda|q|q\frac{1}{S}e^{Sx} + \Lambda|q|q\frac{1}{S} + p_{\text{in}}^2\right)e^{-Sx}.$$

□

Evaluating the solution (2.10) at the endpoint $x = L$ of the pipe and setting $p(L) = p_{\text{out}}$ we finally obtain the well-known relationship of inlet and outlet pressures and the mass flow through the pipe

$$p_{\text{out}}^2 = \left(p_{\text{in}}^2 - \Lambda L|q|q\frac{e^{SL}-1}{SL}\right)e^{-SL}. \tag{2.12}$$

Note that equation (2.12) is not defined for horizontal pipes with a slope equal to zero. In this case we can either solve the trivial ODE (2.11) with $S = 0$ or taking the limit for $s \to 0$ and hence $S \to 0$ in (2.10) using L'Hôpital's rule.

Theorem 2.3. *For horizontal pipes with a slope of $\partial_x h = s = 0$, the solution $p(x)$ to (2.9) with the initial value $p(0) = p_{in}$ is given by*

$$p(x)^2 = p_{in}^2 - x\Lambda|q|q \tag{2.13}$$

with Λ as in (2.10).

As above we evaluate solution (2.13) at $x = L$ and obtain the pressure loss formula for horizontal pipes

$$p_{\text{out}}^2 = p_{\text{in}}^2 - \Lambda L|q|q.$$

Now we need to find a good way to approximate the mean values z_m and T_m. A common choice for the mean temperature is the average temperature

$$T_m := \frac{1}{2}(T_{\text{in}} + T_{\text{out}}).$$

This formula has the advantage that no energy equation needs to be involved. The compressibility factor $z_m = z(p_m, T_m)$ is either defined by the AGA Formula (2.6) or Papay's Formula (2.7), so we need an adequate mean value p_m.

Lemma 2.4. *Let $p(x)$ be given as in (2.13), and let*

$$p_m := \frac{1}{L}\int_0^L p(x)\,\mathrm{d}x$$

be the mean pressure along the pipe. Then

$$p_m = \frac{2}{3}\left(p_{in} + p_{out} - \frac{p_{in}\,p_{out}}{p_{in}+p_{out}}\right). \tag{2.14}$$

Proof. Using equation (2.13) and equation (2.12) we can find a closed formula for $p(x)$ independet from the flow q and any mean values

$$p(x) = \sqrt{p_{\text{in}}^2 - \frac{x}{L}\left(p_{\text{in}}^2 - p_{\text{out}}^2\right)}.$$

Thereby, solving the integral yields the desired formula (2.14).

$$\begin{aligned} p_m &= \frac{1}{L}\int_0^L \sqrt{p_{\text{in}}^2 - \frac{x}{L}\left(p_{\text{in}}^2 - p_{\text{out}}^2\right)}\,\mathrm{d}x \\ &= \frac{2}{3L}\left(\frac{(p_{\text{in}}^2 - \frac{L}{L}(p_{\text{in}}^2 - p_{\text{out}}^2))^{3/2} - (p_{\text{in}}^2 - 0)^{3/2}}{-\frac{1}{L}(p_{\text{in}}^2 - p_{\text{out}}^2)}\right) \\ &= \frac{2}{3}\left(p_{\text{in}} + p_{\text{out}} - \frac{p_{\text{in}}\,p_{\text{out}}}{p_{\text{in}}+p_{\text{out}}}\right). \end{aligned}$$

□

This formula for p_m only depends on p_{in} and p_{out} and yields better results than a simple arithmetic mean.

2.4 The Network Representation

Up to now we only discussed how to model single network elements. The complete network for gas transport is modeled by a directed graph $\mathcal{G} = (V, E)$ consisting of nodes V and edges E. Here edges $e \in E$ represent pipes, valves, resistors, heaters, or regulators. Note that G can contain loops but no self-loops with $e = (v_i, v_i)$.
The set of nodes V is separated into the subsets of supply nodes V_+ with given pressure, the subset of demand nodes V_- with given mass flow and the set of inner nodes V_0

$$V = V_+ \cup V_- \cup V_0.$$

As usual in graph theory the incoming edges of some node v_j are denoted by $\delta_j^- := \{e = (v_i, v_j) \in E\}$ and the outgoing edges by $\delta_j^+ := \{e = (v_j, v_k) \in E\}$, respectively. The set of all incident edges is $\delta_j := \delta_j^- \cup \delta_j^+$.

2.4.1 The System of Equations

In the following we assume that the gas network consists of horizontal pipes only. We study the static case because the gas dynamics change very slowly in time and for most questions it is not necessary to consider the transient case.

The Nodes

For each node we obtain one equation depending one the type of node. At inner nodes where the pipes are connected, the principle of mass conservation – corresponding to Kirchhoff's law – must be fulfilled:

$$\sum_{v_i \in \delta_j^-} q_{ij} - \sum_{v_k \in \delta_j^+} q_{jk} = 0 \qquad \forall v_j \in V_0.$$

At demand nodes the extracted mass flow is preset

$$\sum_{v_i \in \delta_j^-} q_{ij} - \sum_{v_k \in \delta_j^+} q_{jk} = d_j \qquad \forall v_j \in V_-,$$

whereas at supply nodes the inlet pressure is given

$$p_i = s_i \qquad \forall v_i \in V_+.$$

The Edges

In addition to the equations for the nodes, the systems of equations contains also one equation per edge. Assuming a constant compressibility factor $z_{ij} = (z_i + z_j)/2$ computed with Papay's formula (2.7), we can use Theorem 2.3 to calculate the pressure drop along pipe e_{ij} with

$$p_j^2 - p_i^2 = \Lambda_{ij} L_{ij} |q_{ij}| q_{ij}.$$

The friction coefficient is determined with the more accurate formula of Hofer (2.3).

Regulators are modeled with the equation

$$\max(\min(\min(\min(p_i - p_l, -p_j + p_h), -q_{ij} + q_h), p_i - p_j), -q_{ij}) = 0.$$

The equations for other types of elements are not mentioned here because the considered gas network only consists of pipes and regulators. For simulating the gas flow in such a network we used the simulator MYNTS [CCH+16]. A solution of the gas network consists of the pressure p_i and the density ρ_i at nodes, and the mass flow q_{ij} in pipes.

3 Uncertainty Quantification

In this chapter methods for uncertainty quantification (UQ) are introduced following [Sul15]. But first, what is uncertainty quantification? In many applications from engineering and science, uncertainties arise in input data, e.g. in geometry, boundary conditions, or model parameters. It is common to distinguish between two types of uncertainty, epistemic and aleatory uncertainty. We call an uncertainty systematic or epistemic – from the Greek word *ἐπιστὴμη*, meaning knowledge – if the variable has a certain value that could be known in principle but is not in practice, e.g. the soil temperature at a particular time and place. Increasing the number of measurements would lead to a reduction of uncertainty. In contrast, this reduction is not possible for statistical or aleatory uncertainties – from the Latin word *alea*, meaning dice – because in this case the variable does not have a certain value but is random, e.g. if the variable represents a noisy signal that differs each time we run the same experiment. In real life applications, both types of uncertainties are present, so uncertainty quantification must be able to handle both.

In order to describe different problems in the context of uncertainty quantification, suppose we have an input X in some space $\mathcal{X}$ that is mapped by a system F to outputs Y in some space $\mathcal{Y}$. Then some objectives in the context of uncertainty quantification are:

The Forward Propagation Problem. This is the classical question in uncertainty quantification: How does the uncertain or random input affect the system's output? Or in a more mathematical formulation: suppose the uncertainties in the input can be characterized by a probability distribution μ on $\mathcal{X}$. How does the induced probability distribution $(F_*\mu)(E) := \mathbb{P}[F(X) \in E]$ look like on the output space $\mathcal{Y}$? Because $(F_*\mu)$ is a high-dimensional object, one often identifies some specific outcomes or quantities of interest (QoI). Such problems arise, for example, in uncertainty quantification studies for groundwater flow in porous media. The uncertain conductivity is modeled with a random diffusion coefficient, see [CQ15, CGP17, BD17, LZ07, ZL04, GKW$^+$07]. In this application the quantities of interest are typically the mean and the variance of pressure and flux.

The Reliability Problem. Suppose we have a failure set $\mathcal{Y}_{\text{fail}}$ where the system's outcome $F(X) \in \mathcal{Y}_{\text{fail}}$ is somehow unacceptable. How large is the failure probability $\mathbb{P}[F(X) \in \mathcal{Y}_{\text{fail}}]$? For example, consider a pressure control valve in a gas network which is only able to handle a certain amount of gas flow. Depending on the uncertain gas withdrawal at multiple demand nodes, how likely is it that the gas flow will exceed the prescribed limit of the control valve?

The Certification or Prediction Problem. Dually to the reliability problem, given a maximum acceptable probability of error $\varepsilon > 0$, find a set $Y_\varepsilon \subset \mathcal{Y}$ such that $\mathbb{P}[F(X) \in Y_\varepsilon] \geq 1 - \varepsilon$, i.e. the prediction $F(X) \in Y_\varepsilon$ is wrong with probability at most ε. This objective is present in structural engineering, see [Duc05, Men97, INC01, Lar93]. For example, one is interested in the probability of a bridge damage caused by ship collision, wind loading, highway loading, ground shaking, liquefaction, and land sliding. In general, the maximum acceptable probability of the failure of a building, due to any cause, is $\varepsilon = 10^{-4} K_s n_d / n_r$, where n_d is the design life (in years), n_r is the number of people at risk in the event of failure, and K_s is a constant depending on the type of building, e.g. $K_s = 0.5$/year for bridges.

The Inverse Problem. Given some observations of the output Y, one attempts to determine the corresponding uncertain inputs X such that $F(X) = Y$. It is often the case that a computational model requires physical observations to adjust model parameters, initial conditions, and/or boundary conditions. In a typical inverse problem these quantities are determined by minimizing the discrepancy between physical observations and computational model output. This discrepancy between observations can be formalized into a likelihood function which is produced from a probability model for the data, given the model parameters. One application is e.g. haemodynamics, where the material properties of the arterial wall in a segment of an artery are identified by measuring the blood inflow and the pressure drop over the segment, see [RMR88, QBS$^+$06, PVV11, LMQR13].

The Model Reduction Problem. Construct another function F_h (perhaps a numerical model with certain numerical parameters to be calibrated, or one involving fewer input or output variables) such that $F_h \approx F$ in an appropriate sense. Quantifying the accuracy of the approximation may itself be a reliability or prediction problem. For example, consider the simulation of a car crash test [CCMHD16, Dud08, GF16, LGBD$^+$18]. The car model has up to several million nodes and degrees of freedom which results in an enormous amount of data to be handled during the simulation. In a front impact crash test simulation at low-speed only a small part in the front of the car underlies nonlinear plastic deformation. The rear part of the car, which is away from the impactor, deforms only in the linear elastic range. Hence, that part of the car model can be reduced to a limited number of degrees of freedom without losing accuracy.

The first section of this chapter lays out basic concepts of probability theory which are essential for the description of uncertainty quantification problems. The second section gives an overview on different methods for numerical integration. All these

methods have not been specifically developed for uncertainty quantification but are often used in this context for the calculation of expectations. Both sections can be skipped by an advanced reader. Only the last two sections are concerned with mathematical tools that are much closer to the practice of uncertainty quantification. Section 3.3 introduces spectral decompositions of random variables and two different approaches – an intrusive and an non-intrusive one – for determination of spectral expansion coefficients. Finally, Section 3.4 covers the alternative sample-based and, hence, non-intrusive method of stochastic collocation.

3.1 Introduction to Probability Theory

Probability theory is essential for uncertainty quantification because uncertain variables are modeled with random variables or random fields. This is the motivation for providing some important background information on probability theory as in Chapter 2 of [KF09].

3.1.1 Probability Distributions

Probability theory deals with the formal foundations for discussing the degree of confidence that an uncertain event will occur. First of all, we need to define to which events we want to assign a probability. The non-empty set Ω of all possible outcomes or results of an experiment is called *sample space* and a subset $A \subset \Omega$ is called *event*. The *event space*, the set of all measurable events $\mathcal{A}$ to which we are willing to assign probabilities, must be a σ-algebra.

Definition 3.1 (σ-Algebra)**.** Let Ω be a non-empty set. The subset $\mathcal{A}(\Omega)$ of the power set $\mathcal{P}(\Omega)$ is called *σ-algebra* if and only if the following properties hold:

1. It contains the empty set: $\emptyset \in \mathcal{A}(\Omega)$.
2. It is closed under complementation: $A \in \mathcal{A}(\Omega) \Rightarrow A^c = \Omega \setminus A \in \mathcal{A}(\Omega)$.
3. It is closed under countable unions: $A_1, A_2, A_3, \ldots \in \mathcal{A}(\Omega) \Rightarrow \bigcup_{i=1}^{\infty} A_i \in \mathcal{A}(\Omega)$.

Thus, $\mathcal{A}_1 = \{\emptyset, \Omega\}$ is the smallest possible σ-algebra on Ω and the power set $\mathcal{A}_2 = \mathcal{P}(\Omega)$ is the largest possible one. For any subset $A \subset \Omega$, $\mathcal{A}_3 = \{\emptyset, A, A^c, \Omega\}$ is the smallest σ-algebra containing A.

With the help of the Kolmogorov axioms [Kol33] we can define a probability measure $\mathbb{P}$ on $\mathcal{A}(\Omega)$ which assigns a probability to each event.

Definition 3.2 (Probability Distribution)**.** Let Ω be a sample space and $\mathcal{A}(\Omega)$ the corresponding σ-algebra. A probability distribution $\mathbb{P}$ over $(\Omega, \mathcal{A}(\Omega))$ is a mapping of events in the σ-algebra to real values that satisfies the following conditions:

1. Probabilities are not negative: $\mathbb{P}(A) \geq 0$, $\forall A \in \mathcal{A}(\Omega)$.
2. The trivial event Ω of all possible outcomes has the maximum probability of 1: $\mathbb{P}(\Omega) = 1$.
3. It is σ-additive: the probability of countable pairwise disjoint events $A_1, A_2, A_3, \ldots \in \mathcal{A}(\Omega)$ can be written as the sum of the probabilities of each event

$$\mathbb{P}\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mathbb{P}(A_i).$$

Depending on the concrete sample space we distinguish between different types of probability distributions. A discrete probability distribution can be described by a discrete list of the probabilities of outcomes, e.g. the outcomes of rolling a dice. On the other hand, if a set of possible outcomes takes on values in a continuous range, such as the height of a person, the probability distribution is continuous and the probability of any individual outcome equals 0. A probability distribution whose sample space is the set of real numbers is called univariate, while a distribution whose sample space is a vector space is called multivariate.

The collection $(\Omega, \mathcal{A}, \mathbb{P})$ of a sample space Ω, the corresponding σ-algebra $\mathcal{A}(\Omega)$ and a probability measure $\mathbb{P} : \mathcal{A} \to [0, 1]$ is called *probability space*.

3.1.2 Random Variables

Our definition of probability distributions was based on events. Usually one is not interested in the complete event but only some specific attribute of an outcome. A random variable is a function which assigns a (real-valued) value to each outcome of a random experiment. These values are called realizations of the random variable.

Definition 3.3 (Random Variable)**.** Let $(\Omega, \mathcal{A}, \mathbb{P})$ be a probability space, then the function $X : \Omega \to \mathbb{R}$ is called real-valued random variable on Ω if it holds:

$$\forall x \in \mathbb{R} : \ \{\omega \,|\, X(\omega) \leq x\} \in \mathcal{A}.$$

This means that the set of all outcomes whose realizations are less than or equal to a certain value x, must be an event of the σ-algebra $\mathcal{A}$.

It is common to neglect a random variable's dependence on ω. So, in the following we denote a random variable by X instead of $X(\omega)$ unless it is important to mention the dependence. The next question is how to define probability distributions over continuous random variables. Usually they are defined by integrating a probability density function (PDF).

Definition 3.4 (Density Function)**.** Let $\rho : \mathbb{R} \to \mathbb{R}$ be a function with the following properties:

1. ρ is non-negative: $\rho(x) \geq 0 \quad \forall x \in \mathbb{R}$.
2. ρ is integrable.
3. ρ is normalized: $\int_{-\infty}^{\infty} f(x)\,\mathrm{d}x = 1$.

Then ρ is called probability density function and defines by

$$\mathbb{P}[X \leq a] = \int_{-\infty}^{a} \rho(x)\,\mathrm{d}x$$

a probability distribution over the real numbers. The function $\mathbb{P}$ is the *cumulative density distribution* (CDF) of X.

The simplest probability density function is the uniform distribution where all intervals of the same length on the distribution's support are equally likely.

Definition 3.5 (Uniform Distribution)**.** A random variable X has a uniform distribution over $[a, b]$, denoted by $X \sim \mathcal{U}([a, b])$ if it has the probability density function

$$\rho(x) = \begin{cases} \frac{1}{b-a} & a \leq x \leq b, \\ 0 & \text{otherwise.} \end{cases}$$

A random variable with uniform distribution has a mean of $\mathbb{E}[X] = \frac{1}{2}(a + b)$ and a variance of $\mathrm{Var}[X] = \frac{1}{12}(b - a)^2$. Another often used probability density function is the more complex Gaussian distribution.

Definition 3.6 (Gaussian Distribution)**.** A random variable X has a Gaussian distribution with mean μ and variance σ^2, denoted by $X \sim \mathcal{N}(\mu, \sigma^2)$, if it has the probability density function

$$\rho(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right).$$

The standard Gaussian distribution has a mean of 0 and a variance of 1. A Gaussian distribution is a bell-shaped curve, where the mean μ determines the location of the peak and the variance σ^2 the width of the peak. The smaller the variance is the narrower and higher is the peak. Figure 3.1 shows the probability density functions of four different Gaussian distributions.

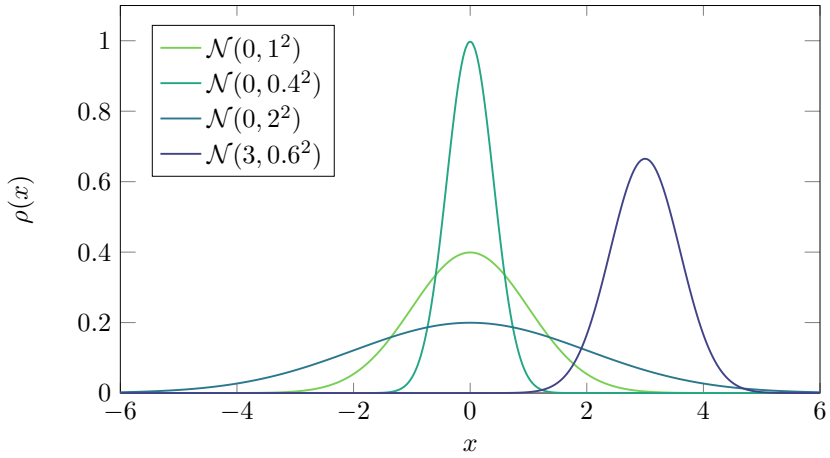


Figure 3.1: Example PDFs of four Gaussian distributions.

The *joint probability distribution* over multiple random variables $X_1, X_2, \ldots, X_n$ is a probability distribution that gives the probability that each of X_i falls in any particular range or discrete set of values specified for that variable. As in the univariate case, the joint probability distribution can be expressed in terms of a joint probability density function.

Definition 3.7 (Joint Density Function)**.** Let $\rho : \mathbb{R}^n \to \mathbb{R}$ be a function fulfilling the following properties:

1. ρ is non-negative: $\rho(x_1, x_2, \ldots, x_n) \geq 0 \quad \forall x \in \mathbb{R}^n$.
2. ρ is integrable.
3. ρ is normalized: $\int_{-\infty}^{\infty} \ldots \int_{-\infty}^{\infty} \rho(x_1, \ldots, x_n) \, \mathrm{d}x_1 \ldots \, \mathrm{d}x_n = 1$.

Then ρ is called joint probability density function and defines by

$$\mathbb{P}[X_1 \leq a_1, \ldots, X_n \leq a_n] = \int_{-\infty}^{a_1} \ldots \int_{-\infty}^{a_n} \rho(x_1, \ldots, x_n) \, \mathrm{d}x_1 \ldots \, \mathrm{d}x_n \tag{3.1}$$

a joint probability distribution over the real numbers. The function $\mathbb{P}$ is the *joint cumulative density distribution* of $X_1, \ldots, X_n$.

The joint density function can be used to find the *marginal distribution* of any variable integrating out the other variables. If, for example, $\rho(x, y, z)$ is the joint density of X, Y, and Z, then

$$\rho(x) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \rho(x, y, z) \, \mathrm{d}y \, \mathrm{d}z$$

is the marginal density of X.

Definition 3.8 (Independence)**.** Two random variables X and Y are (stochastically) independent if the joint distribution can be written as a product of the marginal distributions

$$\mathbb{P}[X, Y] = \mathbb{P}[X]\,\mathbb{P}[Y]$$

or, equivalently, the joint density

$$\rho(x, y) = \rho(x)\,\rho(y).$$

A collection of random variables is *independent and identically distributed* (iid) if each random variable has the same probability distribution as the others and all are mutually independent.

In the context of uncertainty quantification one often needs to draw random points from a given distribution. Since standard libraries usually only contain functions to generate uniform random points, the following theorem is useful to calculate density functions of transformed random variables.

Theorem 3.9. *Let $X \in \mathbb{R}^d$ be a random variable with density function $\rho : \mathbb{R}^d \to \mathbb{R}_+$ and $\phi : \mathbb{R}^d \to \mathbb{R}^d$ a continuously differentiable transformation with continuously differentiable inverse mapping ϕ^{-1}. Then, the density μ of $Y = \phi(X)$ fulfills*

$$\mu(y) = \rho(\phi^{-1}(y))\,|\det J_{\phi^{-1}}| \tag{3.2}$$

where $J_{\phi^{-1}}$ denotes the Jacobi matrix of the inverse mapping.

Proof. Since $X = \phi^{-1}(Y)$ the following statement is true for any set $A \subset \mathbb{R}^d$:

$$\int_A \mu(y)\,\mathrm{d}y = \mathbb{P}(Y \in A) = \mathbb{P}(X \in \phi^{-1}(A)) = \int_{\phi^{-1}(A)} \rho(x)\,\mathrm{d}x.$$

As ϕ and ϕ^{-1} are continuously differentiable, we can apply the transformation formula from multivariate integral calculus:

$$\int_{\phi^{-1}(A)} \rho(x)\,\mathrm{d}x = \int_A \rho(\phi^{-1}(y))\,|\det J_{\phi^{-1}}|\,\mathrm{d}y.$$

Because this relation holds true for any subset A, it follows that

$$\mu(y) = \rho(\phi^{-1}(y))\,|\det J_{\phi^{-1}}|.$$

□

3.1.3 The Expectation, Variance, and Median

The expected value or the expectation of a random variable is an important statistical value of it and describes the value which the random variable takes on average.

Definition 3.10 (Expectation)**.** Let X be a discrete random variable, then the expectation of X under the distribution $\mathbb{P}$ is

$$\mathbb{E}[X] = \sum_x x \cdot \mathbb{P}[x].$$

Let X be a continuous random variable with density function ρ, then the expectation of X is calculated as

$$\mathbb{E}[X] = \int_{\mathbb{R}} x \ \rho(x) \, \mathrm{d}x.$$

The expected value fulfills some fundamental properties:

1. Linearity: For any two random variables X_1 and X_2, and any real numbers a and b it holds that

   $$\mathbb{E}[aX_1 + bX_2] = a\mathbb{E}[X_1] + b\mathbb{E}[X_2].$$

   Note that this identity is true even if the random variables are not independent.

2. Monotony: If $X \leq Y$ almost sure and if $\mathbb{E}[X]$ and $\mathbb{E}[Y]$ exist, then

   $$\mathbb{E}[X] \leq \mathbb{E}[Y].$$

3. Product of random variables: If X and Y are independent, then

   $$\mathbb{E}[X \cdot Y] = \mathbb{E}[X] \cdot \mathbb{E}[Y].$$

The variance of a random variable is an important measure of dispersion. It describes the expected squared deviation from the mean of the random variable.

Definition 3.11 (Variance)**.** Let X be a random variable with the mean $\mathbb{E}[X]$, then the variance of X is defined as

$$\mathrm{Var}[X] = \mathbb{E}\left[(X - \mathbb{E}[X])^2\right].$$

1. The expression for the variance can be alternatively formulated

   $$\mathrm{Var}[X] = \mathbb{E}[X^2] - \mathbb{E}[X]^2.$$

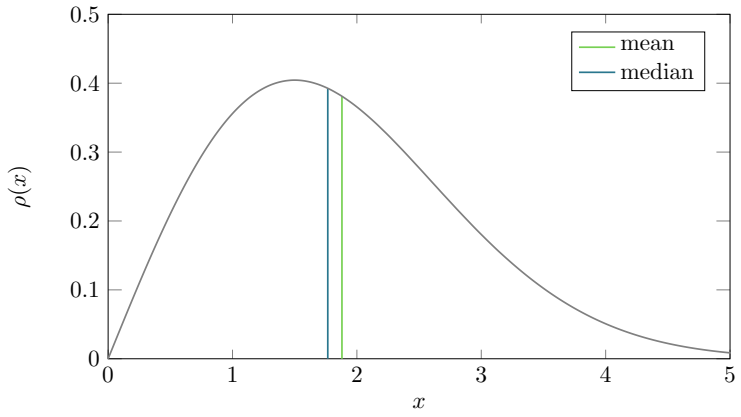


Figure 3.2: The median and mean for a skewed distribution.

2. Linear transformation: for any real numbers a and b we obtain

$$\mathrm{Var}[aX + b] = a^2\mathrm{Var}[X].$$

3. If X and Y are independent, then

$$\mathrm{Var}[X + Y] = \mathrm{Var}[X] + \mathrm{Var}[Y].$$

Another statistical value for the density of random variables is the so-called median. Like the expectation it describes the 'center' of the density, but rare outliers are nearly of no consequence.

Definition 3.12 (Median)**.** Let X be a continuous random variable with density distribution $\mathbb{P}$, then the value m is called a median of X, if the following equations are fulfilled:

$$\mathbb{P}[X \leq m] \geq \frac{1}{2} \quad \text{and} \quad \mathbb{P}[m \leq X] \geq \frac{1}{2}.$$

The median is not mandatorily unique for continuous distributions. Only in cases where the density distribution is strictly increasing in a neighborhood where it is $\frac{1}{2}$, the median is unique.
Figure 3.2 depicts the median and mean of a skewed distribution. For symmetric distributions the median and mean are identical.

Although not all distributions have such a strong loss in the probability of outcomes far away from the mean like the Gaussian distribution, it is possible to quantify the decrease for arbitrary distributions. The following Chebyshev's inequality [Che67] states that no more than $1/k^2$ of the distribution's values can be more than k standard deviations away from the mean μ.

Theorem 3.13 (Chebyshev Inequality)**.** *Let X be a random variable with the mean $\mu := \mathbb{E}$ and a finite variance of $\sigma^2 := \mathrm{Var}[X]$, then for all real numbers $k > 0$ the following inequality holds true:*

$$\mathbb{P}\left[|X - \mu| \geq k\sigma\right] \leq \frac{1}{k^2}. \tag{3.3}$$

Proof. For any event A, let $\mathbb{1}_A$ be the indicator random variable of A, i.e. $\mathbb{1}_A$ equals 1 if A occurs and equals 0 otherwise. Then

$$\begin{aligned}
\mathbb{P}[|X - \mu| \geq k\sigma] &= \mathbb{P}\left[\left(\frac{X-\mu}{k\sigma}\right)^2 \geq 1\right] \\
&= \mathbb{E}\left[\mathbb{1}_{\left(\frac{X-\mu}{k\sigma}\right)^2 \geq 1}\right] \\
&\leq \mathbb{E}\left[\left(\frac{X-\mu}{k\sigma}\right)^2\right] \\
&= \frac{\mathbb{E}[(X-\mu)^2]}{(k\sigma)^2} \\
&= \frac{1}{k^2}
\end{aligned}$$

□

This proof shows why the bounds are not sharp in typical cases: from the second to the third line, the value 1 of the indicator function is replaced by $(\frac{X-\mu}{k\sigma})^2$ whenever the latter exceeds 1. Of course, in some cases it exceeds 1 by a very wide margin.

3.2 Numerical Integration

In this section we discuss several methods for the numerical integration of definite integrals. Common objectives in UQ are to compute simple statistics of the solution as expected values or variances, see [AV13, AGP+08, CGST11, CGP17]. These statics are nothing else than Lebesgue integrals with respect to a given measure. The value of such an integral is approximated by evaluating the integrand at a finite number of sampling points. Remember that one function evaluation may correspond to one time-consuming simulation run or one expensive experiment and that the integration domains are usually high-dimensional because practical applications often involve many uncertain parameters. Thus, it is important to have effective methods for accurate numerical integration using as few sample points as possible.
There are three types of quadrature formulas, which differ in the way how sampling points are generated. The classic one is deterministic numerical integration where sampling points are generated deterministically from the measure. In contrast, Monte Carlo integration, which is often used for uncertainty quantification, generates random sampling points from the measure. The last method is something

in between. In quasi-Monte Carlo methods the points are in fact deterministic but somehow look random and distributed accordingly to the measure. All types of numerical integration methods are discussed in the following sections.

3.2.1 The Univariate Quadrature

First, we consider the numerical integration of a real-valued function f with respect to a measure μ over a one-dimensional domain $I \subseteq \mathbb{R}$. The integral is approximated by a weighted sum of f, which is evaluated at predetermined sampling points of I.

Definition 3.14 (Quadrature Formula)**.** Let f be a real-valued function. A quadrature formula $Q(f)$ approximates the integral of f with respect to measure μ over the domain $I \subseteq \mathbb{R}$

$$\int_I f(x)\,\mathrm{d}\mu(x) \approx Q(f) := \sum_{i=1}^{n} w_i f(x_i),$$

with given *nodes* $x_1, x_2, \ldots, x_n \in I$ and *weights* $w_1, w_2, \ldots, w_n \in \mathbb{R}$.

The objective is to choose quadrature nodes and weights in a way that the approximation $\int_I f\,\mathrm{d}\mu \approx Q(f)$ is accurate for a large number of integrands f. One way to measure the accuracy of the approximation is the following:

Definition 3.15 (Order of Accuracy)**.** If the quadrature formula $Q(f)$ is exact for all polynomials $p \in \mathcal{P}_n$ of degree at most $n \in \mathbb{N}_0$, i.e.

$$\int_I p(x)\,\mathrm{d}\mu(x) = Q(p),$$

then Q is said to have an order of accuracy of n.

Newton-Cotes Formulas

In the following let μ be the Lebesgue measure on the interval $I = [a, b]$. The simplest quadrature formula only has one node at the center of I and therefore is called midpoint rule.

Definition 3.16 (Midpoint Rule)**.** The midpoint quadrature formula has a single node $x_1 = a + \frac{b-a}{2}$ with the weight $w_1 = |b - a|$, i.e. the approximation yields

$$\int_a^b f(x)\,\mathrm{d}x \approx Q_1(f) := f\left(a + \frac{b-a}{2}\right)|b-a|. \tag{3.4}$$

The midpoint rule can be interpreted as approximating the integrand f by the constant function with value $f\left(a + \frac{b-a}{2}\right)$ and it is easy to see that this quadrature formula is exact for all linear polynomials, i.e. it has an order of accuracy of 1. Another possibility is to approximate the integrand f by the linear function that equals $f(a)$ at a and $f(b)$ at b. Thus, we obtain the trapezoidal rule with an order of accuracy of 2.

Definition 3.17 (Trapezoidal Rule)**.** The trapezoidal quadrature rule has the nodes $x_1 = a$ and $x_2 = b$, and the weights $w_1 = w_2 = \frac{|b-a|}{2}$, i.e.

$$\int_a^b f(x)\,\mathrm{d}x \approx Q_2(f) := \Big(f(a) + f(b)\Big)\frac{|b-a|}{2}. \tag{3.5}$$

Of course, we can also approximate the integrand f with a polynomial of degree n. For this we choose $n+1$ distinct nodes $x_i \in [a, b]$ which we interpolate with the Lagrange polynomial

$$L_n(x) := \sum_{i=0}^{n} f(x_i)\,\ell_i(x), \tag{3.6}$$

where $\ell_i(x) \in \mathcal{P}_n$ denotes the basis polynomial

$$\ell_i(x) := \prod_{\substack{0\le k\le n\\ k\neq i}} \frac{x - x_k}{x_i - x_k}. \tag{3.7}$$

For equidistant sampling points x_i, we obtain the Newton-Cotes quadrature formulas.

Definition 3.18 (Newton-Cotes Formula)**.** Consider $n+1$ equidistant points $x_i = a + ih$ with $h = \frac{b-a}{n}$. The quadrature formula

$$\int_a^b f(x)\,\mathrm{d}x \approx \int_a^b L_n(x)\,\mathrm{d}x = \sum_{i=0}^{n} w_i f(x_i) =: Q(f) \tag{3.8}$$

that arises from approximating f by the Lagrange polynomial L_n interpolating f at points $\{x_i\}_{i=0,\dots,n}$ is called the closed Newton-Cotes quadrature formula and has the weights

$$w_i = \int_a^b \ell_i(x)\,\mathrm{d}x.$$

If we do not use $x_0 = a$ and $x_n = b$ but only $\{x_i\}_{i=1,\dots,n-1}$ to construct the Lagrange polynomial, the resulting quadrature formula is called the open Newton-Cotes formula.

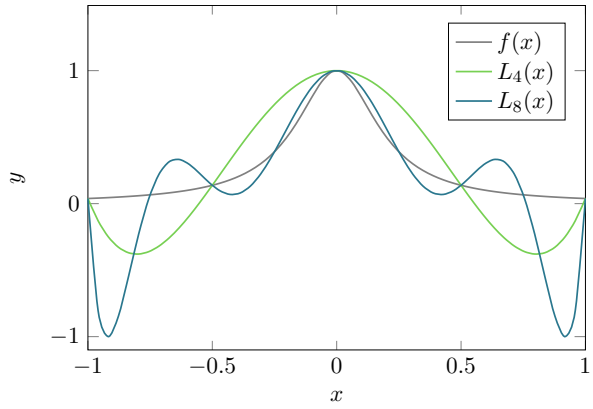


Figure 3.3: The interpolation of the Runge function (gray) with the Lagrange polynomials L_4 (green) and L_8 (blue).

The quadrature rules (3.4) and (3.5) are also Newton-Cotes formulas. The midpoint rule is the open Newton-Cotes formula on the three points $x_1 = a$, $x_2 = a + \frac{b-a}{2}$, and $x_3 = b$, and the trapezoidal rule is the closed Newton-Cotes formula on the two points $x_1 = a$ and $x_2 = b$.

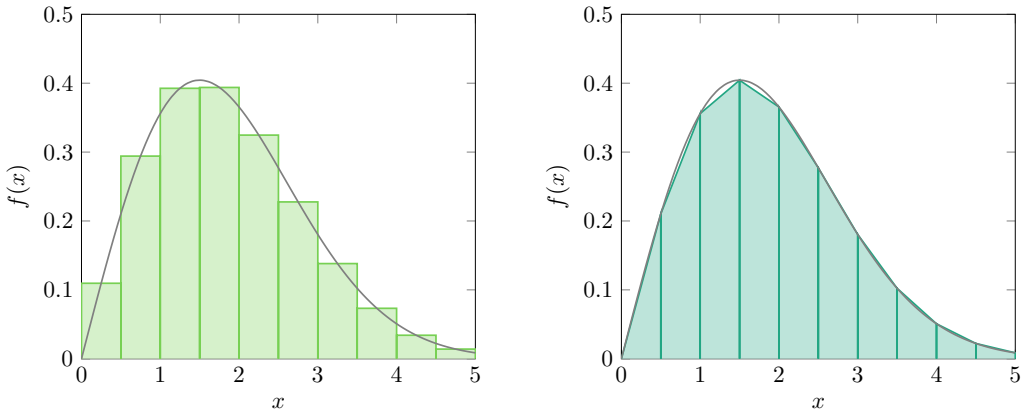
Runge's Phenomenon. Can we improve the accuracy of the Newton-Cotes formula by using more sampling points? Not in general. Higher-order Lagrange polynomials are not necessarily similar to the integrand, because polynomials tend to $\pm\infty$ in the limit $x \to \pm\infty$. If the integrand is periodic or asymptotically constant, large oscillations will occur near the boundary. This phenomenon is known as Runge's phenomenon. Consider the Runge function

$$f(x) = \frac{1}{1 + 25x^2},$$

which tends to 0 for $x \to \pm\infty$. In Figure 3.3 the Runge function and the Lagrange polynomials $L_4(x)$ and $L_8(x)$ are plotted for equidistant sampling points $x_i := \frac{2i}{n} - 1$. At the interpolating points, the error between the function and the Lagrange polynomial is zero by definition. Between the sampling points, especially near the boundary, the error between the function and the interpolating polynomial increases for higher-order polynomials. For these types of functions, Newton-Cotes formulas based on polynomial interpolation over the complete integration domain are inappropriate.

The Riemann Sum. To overcome this phenomenon, the quadrature over $[a, b]$ is often done by taking a uniform or non-uniform partition of the interval $[a, b]$ and applying a simple quadrature rule to each subinterval. These types of quadrature rules are called *Riemann sum* quadrature rules. Consider the uniform partition

$$p_0 = a, \quad p_1 = a + h, \quad \dots \quad , \quad p_n = a + nh = b$$



(a) The midpoint rule.

(b) The trapezoidal rule.

Figure 3.4: The Riemann sum quadrature with the midpoint rule (a) and the trapezoidal rule (b).

with $h = \frac{b-a}{n}$. Applying the midpoint rule to each subinterval yields

$$\int_a^b f(x)\,\mathrm{d}x \approx \sum_{i=0}^{n-1} hf\left(a + \frac{h}{2} + ih\right) =: Q(f), \tag{3.9}$$

which can be interpreted as integration by a piecewise constant approximation. Applying the trapezoidal rule to each subinterval is equivalent to a piecewise linear approximation and yields

$$\int_a^b f(x)\,\mathrm{d}x \approx \sum_{i=0}^{n-1} \frac{h}{2}\Big(f(a+ih) + f(a+(i+1)h)\Big) =: Q(f). \tag{3.10}$$

See Figure 3.4 for an illustration of a Riemann sum quadrature based on the midpoint rule (left) and the trapezoidal rule (right) with ten subintervals respectively. Note that rules (3.9) and (3.10) are not n-point Newton-Cotes formulas.

The Gaussian Quadrature

Another group of quadrature formula are Gaussian quadrature formulas in which both the nodes as well as the weights are optimally chosen in the sense that the order of accuracy is maximal. With the optimal choice of n nodes and weights the Gaussian quadrature is exact for polynomials of degree $2n-1$. Furthermore, all weights are positive so that the quadrature formula is stable for a large number n of quadrature points. Recall that we want to approximate a definite integral with respect to a measure μ

$$\int_a^b f(x)\,\mathrm{d}\mu(x) \approx \sum_{i=1}^{n} w_i f(x_i) =: Q(f).$$

For Gaussian quadrature, let $\{q_n, n \in \mathbb{N}\}$ be a system of orthogonal polynomials for μ, i.e. q_n is a polynomial of degree n such that

$$\int_a^b p(x)q_n(x)\,\mathrm{d}\mu(x) = 0 \quad \forall p \in \mathcal{P}_{n-1}.$$

Since the orthogonal polynomial q_n has n distinct roots in $[a, b]$, we will use the zeros $x_1, \ldots, x_n$ as quadrature points.

Definition 3.19 (Gauss Quadrature)**.** The n-point Gauss quadrature formula Q_n is the quadrature formula with nodes $x_1, \ldots, x_n$ given by the zeros of the orthogonal polynomial q_n and weights given in terms of the corresponding Lagrange basis polynomials

$$w_i := \int_a^b l_i(x)\,\mathrm{d}\mu(x) = \int_a^b \prod_{\substack{1\le j\le n\\ j\neq i}} \frac{x - x_j}{x_i - x_j}\,\mathrm{d}\mu(x).$$

It is easy to see that the n-point Gaussian quadrature is exact for polynomials $p \in \mathcal{P}_{n-1}$ of degree at most $n - 1$. Obviously, it coincides with its Lagrange-form interpolation at nodes $x_1, \ldots, x_n$, i.e.

$$p(x) = \sum_{i=1}^n p(x_i)l_i(x).$$

Hence,

$$\int_a^b p(x)\,\mathrm{d}\mu(x) = \int_a^b \sum_{i=1}^n p(x_i)l_i(x)\,\mathrm{d}\mu(x) = \sum_{i=1}^n p(x_i)w_i =: Q_n(p).$$

Moreover, Gaussian quadrature is optimal in the sense that its degree of polynomial exactness is maximal:

Theorem 3.20. *The n-point Gaussian quadrature formula has an order of accuracy of $2n - 1$ and not any other quadrature formula on n nodes has a higher order of accuracy.*

Proof. Consider $p \in \mathcal{P}_{\le 2n-1}$. Factorizing this polynomial yields

$$p(x) = g(x)q_n(x) + r(x)$$

where $\deg(g) \le n - 1$, and the remainder r is also of degree at most $n - 1$. Since q_n is orthogonal to all polynomials of degree at most $n - 1$, $\int_a^b gq_n\,\mathrm{d}\mu(x) = 0$. In addition, since $g(x_i)q_n(x_i) = 0$ for each quadrature point x_i,

$$Q_n(gq_n) = \sum_{i=1}^n w_i q_n(x_i) = 0.$$

Since both integration as well as quadrature are linear operators, we have

$$\int_a^b p \, \mathrm{d}\mu(x) = \int_a^b r \, \mathrm{d}\mu(x) \text{ and } Q_n(p) = Q_n(r).$$

Because r has a polynomial degree of at most $n-1$, the quadrature formula is exact and hence

$$\int_a^b p \, \mathrm{d}\mu(x) = Q_n(p).$$

To show that a quadrature formula on n distinct nodes $x_1, \ldots, x_n$ with any weights w_i cannot have a higher accuracy, consider the polynomial $f(x) := \prod_{j=1}^n (x - x_j)^2$. Then

$$\int_a^b f(x) \, \mathrm{d}\mu(x) > 0 = \sum_{i=1}^n w_i f(x_i),$$

since f vanishes at each node x_i. Hence, the quadrature formula is not exact for polynomials of degree $2n$. □

As mentioned above, a further advantage of Gaussian quadrature is that the weights are positive:

Theorem 3.21. *For any non-negative measure μ on $\mathbb{R}$, the Gauss quadrature weights are positive.*

Proof. Consider the polynomial

$$p(x) := \prod_{\substack{1 \le j \le n \\ j \ne i}} (x - x_j)^2$$

for fixed $1 \le i \le n$. Since p has a polynomial degree smaller than $2n-1$, the Gauss quadrature is exact, and since p vanishes at every node $x_j \ne x_i$, it follows

$$\int_a^b p(x) \, \mathrm{d}\mu(x) = \sum_{j=1}^n w_j p(x_j) = w_i p(x_i).$$

Since μ is a non-negative measure, $p \ge 0$ everywhere, and $p(x_i) > 0$, it follows that $w_i > 0$. □

The next theorem quantifies the error made by Gaussian quadrature for non-polynomial integrands, cf. [Sto06].

Theorem 3.22. *Suppose that $f \in \mathcal{C}^{2n}([a,b])$. Then there exists $\xi \in [a,b]$ such that*

$$\int_a^b f(x)\,\mathrm{d}\mu(x) - Q_n(f) = \frac{f^{(2n)}(\xi)}{(2n)!}\|p_n\|_{L^2(\mu)},$$

where p_n is the monic orthogonal polynomial of degree n for μ. In particular,

$$\left|\int_a^b f(x)\,\mathrm{d}\mu(x) - Q_n(f)\right| \leq \frac{\|f^{(2n)}\|}{(2n)!}\|p_n\|_{L^2(\mu)},$$

and the error is zero if f is a polynomial of degree at most $2n-1$.

Despite Gaussian quadrature formulas having an optimal order of accuracy, they are not always the first choice. One drawback is the computational cost of $\mathcal{O}(n^2)$ for computing the weights. Another drawback is that the nodes are not nested. Therefore, function evaluations in Q_n cannot be reused in a more accurate quadrature formula Q_m with $m > n$. This motivates the introduction of the nested Clenshaw-Curtis quadrature rules in the next section.

In the context of uncertainty quantification it is not common to use Gaussian quadrature formulas to compute expected values or other moments of the solution. The reason is that, typically, only the distribution of uncertain input variables is known and not the distribution of the propagated uncertainty in the solution. Thus, it is impossible to compute optimal quadrature points and weights for the quantities of interest. But nevertheless, Gaussian quadrature formulas are used for uncertainty quantification, in particular for calculating the coefficients of a polynomial chaos expansion, see [KW16, TI14, AV13]. The idea of a polynomial chaos expansion is to expand an uncertain input variable with a density ρ in terms of the corresponding orthogonal polynomials. Further details can be found in Section 3.3.2. Each coefficient of the expansion is calculated by evaluating one integral with respect to ρ. So the Gaussian quadrature nodes and weights for ρ have to be computed once and can then be reused to calculate every coefficient.

The Clenshaw-Curtis Quadrature

The Clenshaw-Curtis quadrature rule is a very often used quadrature formula which not only avoids large oscillations near the boundary because more quadrature nodes are placed there, but also has the advantage of nested points. If the quadrature nodes are nested, the first function evaluations or simulation runs can be reused for more accurate quadrature formulas. Moreover, nested quadrature rules are the basis for multivariate sparse grid quadrature formulas in the next section.

For the definition of the Clenshaw-Curtis quadrature formula [CC60] consider an integration over the interval $I = [-1,1]$ with respect to the Lebesgue measure. We

start with a change of variables:

$$\int_{-1}^{1} f(x)\,\mathrm{d}x = \int_{0}^{\pi} f(\cos\theta)\sin\theta\,\mathrm{d}\theta.$$

That is, we have transformed the problem from integrating $f(x)$ to one of integrating $f(\cos\theta)\sin\theta$. This can be performed if we know the cosine series for $f(\cos\theta)$, namely

$$f(\cos\theta) = \frac{a_0}{2} + \sum_{k=1}^{\infty} a_k \cos(k\theta).$$

Then the integral becomes

$$\int_{0}^{\pi} f(\cos\theta)\sin(\theta)\,d\theta = a_0 + \sum_{k=1}^{\infty} \frac{2a_{2k}}{1-(2k)^2}.$$

In order to calculate the cosine series coefficients

$$a_k = \frac{2}{\pi}\int_{0}^{\pi} f(\cos\theta)\cos(k\theta)\,d\theta,$$

one must again perform a numeric integration, so at first glance this may not seem to have simplified the problem. But by the Nyquist-Shannon sampling theorem [Nyq28, Sha49], for $k \leq n$ the coefficients a_k can be computed exactly by evaluating $f(\cos\theta)$ at the $n+1$ equidistant and equally weighted points $\theta_j = \frac{j\pi}{n}$ for $j = 0, \ldots, n$. Only the endpoints are weighted by $1/2$ to avoid double-counting, equivalent to the trapezoidal rule. Hence, we obtain

$$a_k \approx \frac{2}{n}\left[\frac{f(1)}{2} + \sum_{j=1}^{n-1} f(\cos\tfrac{j\pi}{n})\cos\tfrac{kj\pi}{n} + \frac{f(-1)}{2}(-1)^k\right]. \tag{3.11}$$

For larger $k > n$, formula (3.11) is wrong and because of aliasing, one only computes the coefficients a_{2k} up to $k = \lfloor \frac{n}{2} \rfloor$, since discrete sampling of the function makes the frequency of $2k$ indistinguishable from that of $n - 2k$. Hence, the Clenshaw-Curtis quadrature formula is given by

$$\int_{0}^{\pi} f(\cos\theta)\sin(\theta)\,\mathrm{d}\theta \approx a_0 + \sum_{k=1}^{\lfloor n/2 \rfloor} \frac{2a_{2k}}{1-(2k)^2}. \tag{3.12}$$

Note that the cosine series expansion of f is also an approximation of f by Chebyshev polynomials, because by definition $T_k(\cos\theta) = \cos(k\theta)$:

$$f(x) = \frac{a_0}{2}T_0(x) + \sum_{k=1}^{\infty} a_k T_k(x),$$

and thus we integrate $f(x)$ by integrating its approximate expansion in terms of Chebyshev polynomials. The nodes $x_j = \cos\frac{j\pi}{n}$ correspond to the extrema of the Chebyshev polynomial $T_n(x)$.

In the previous section we have seen that the n Gaussian quadrature nodes are constructed such that they integrate exactly polynomials up to a degree of $2n-1$. In contrast, Clenshaw-Curtis quadrature evaluates the integrand at $n+1$ points and exactly integrates polynomials only up to a degree of n. Therefore, one might think that Clenshaw-Curtis quadrature is worse than Gaussian quadrature, but in practical applications this is not the case. In fact, Clenshaw-Curtis quadrature can be as accurate as Gaussian quadrature for the same number of points [Tre08, CE09, Nov11], since most numeric integrands are not polynomials and the approximation of many functions in terms of Chebyshev polynomials converges fast [DW81, Boy82]. Due to this fact plus the above discussed advantage of nested points, the Clenshaw-Curtis quadrature rule is the method of choice in many applications.

3.2.2 The Multivariate Quadrature

After discussing numerical methods for integrals over a one-dimensional domain, we now introduce quadrature formulas for multi-dimensional integrals, i.e. integrals of the form

$$\int_\Omega f(\mathbf{x})\,\mathrm{d}\mathbf{x} = \int_{a_d}^{b_d} \dots \int_{a_1}^{b_1} f(x_1,\dots,x_d)\,\mathrm{d}x_1 \dots\,\mathrm{d}x_d,$$

with the integration domain $\Omega = \prod_{i=1}^{d}[a_i, b_i]$.

Tensor Product Quadrature

In general, we do not need a new method for multi-variate quadrature because a d-dimensional integration can be interpreted as a sequence of one-dimensional integrations, i.e. we choose a one-dimensional n-point quadrature formula $Q^1(f)$ and apply it d times:

$$\begin{aligned}\int_\Omega f(\mathbf{x})\,\mathrm{d}\mathbf{x} &\approx (Q^1 \otimes \dots \otimes Q^1)f \\ &= \sum_{i_1=1}^{n} \dots \sum_{i_d=1}^{n} w_{i_1} \dots w_{i_d} f(x_{i_1},\dots,x_{i_d}) \\ &=: Q^d(f)\end{aligned} \tag{3.13}$$

The resulting quadrature formula $Q^d(f)$ is called *tensor product quadrature*. If the polynomial degree of the one-dimensional quadrature rules is sufficiently large then the error of a product method decreases with the number of points n proportional to

$$\varepsilon = \mathcal{O}(n^{-r/d}) \tag{3.14}$$

for all functions f from the space

$$\mathcal{C}^r := \left\{ f : \Omega \to \mathbb{R} : \max_{|s|_1 \le r} \left\| \frac{\partial^{|\mathbf{s}|_1} f}{\partial x_1^{s_1} \dots \partial x_d^{s_d}} \right\|_\infty < \infty \right\}$$

of functions with bounded derivatives up to order r. The error bound (3.14) indicates the positive impact of the smoothness r as well as the negative impact of the dimension d on the convergence rate. In particular one can see that tensor product quadrature formulas have one drawback: if the one-dimensional quadrature formula needs n nodes to achieve a given accuracy, the tensor product formula needs $N = n^d$ nodes for the same accuracy. This approach quickly leads to an infeasible number of integrand evaluations and thus simulation runs. Especially in the context of uncertainty quantification, practical applications often involve high-dimensional integration domains so that we need to develop new techniques to avoid this *curse of dimensionality*.

Sparse Grids

The idea to break down the curse of dimensionality motivates a sparse grid quadrature formula, which involves substantially less than n^d nodes. Using less sampling points impairs the accuracy but the benefit in complexity outweighs the loss in accuracy.
As in [GG98, Gar13, Pfl10] we consider numerical integration over the d-dimensional hypercube $\Omega = [-1, 1]^d$ by a sequence of n_l^d-point quadrature formulas with a level of $l \in \mathbb{N}$ and $n_l^d < n_{l+1}^d$:

$$\int_\Omega f(\mathbf{x})\,\mathrm{d}\mathbf{x} \approx \sum_{i=1}^{n_l^d} w_{l,i} f(\mathbf{x}_{l,i}) =: Q_l^d(f),$$

with nodes $\mathbf{x}_{l,i}$ and weights $w_{l,i}$. We assume that for each $l \in \mathbb{N}$ a one-dimensional quadrature rule Q_l^1 is given and that these rules are nested, i.e. the nodes for Q_l^1 are a subset of those for Q_{l+1}^1. This is the case, if we use for example some open Newton-Cotes quadrature formula (3.8) with $n_l^1 = 2^l - 1$ nodes or the Clenshaw-Curtis quadrature formulas (3.12) with $n_1^1 = 1$ and $n_l^1 = 2^{l-1} + 1, l \geq 2$ nodes, respectively. Note that in both cases $n_l^1 = \mathcal{O}(2^l)$.

Smolyak's Formula. Smolyak [Smo63] considered for his quadrature formula functions with bounded mixed derivatives with an order of r, i.e.

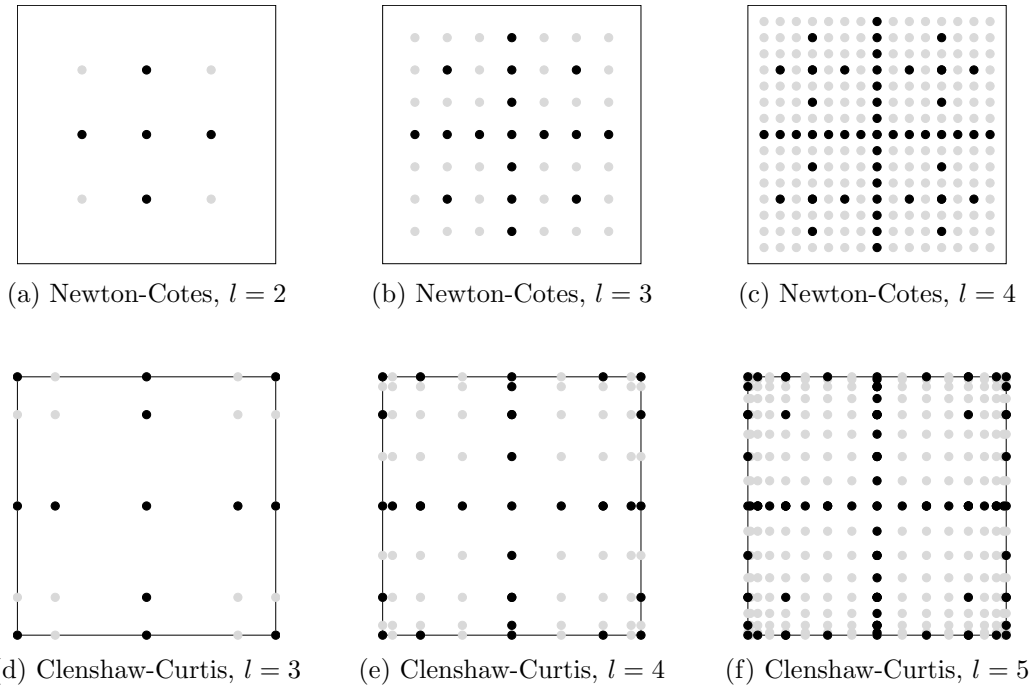
$$\mathcal{W}^r := \left\{ f : \Omega \to \mathbb{R} : \max_{|s|_\infty \leq r} \left\| \frac{\partial^{|\mathbf{s}|_1} f}{\partial x_1^{s_1} \dots \partial x_d^{s_d}} \right\|_\infty < \infty \right\},$$

with $|\mathbf{s}|_1 = s_1 + \ldots + s_d$. For constructing the multi-dimensional quadrature formula, first consider a sequence of one-dimensional quadrature formulas for a univariate function f

$$Q_l^1(f) := \sum_{i=1}^{n_l^1} w_{l,i} f(x_{l,i}).$$

For this sequence we define the difference quadrature formula by

$$\Delta_k f := (Q_k^1 - Q_{k-1}^1) f, \tag{3.15}$$



(a) Newton-Cotes, $l = 2$ (b) Newton-Cotes, $l = 3$ (c) Newton-Cotes, $l = 4$

(d) Clenshaw-Curtis, $l = 3$ (e) Clenshaw-Curtis, $l = 4$ (f) Clenshaw-Curtis, $l = 5$

Figure 3.5: Full grids (gray) and sparse grids (black) of different levels l based on the open Newton-Cotes formula (a)–(c) and the Clenshaw-Curtis formula (d)–(f).

with $Q_0^1 f = 0$, and $Q_1^1 f = 2f(x_0)$. Smolyak's quadrature formula for d-dimensional functions f is defined as

$$Q_l^d := \sum_{|\mathbf{k}|_1 \leq l+d-1} (\Delta_{k_1} \otimes \ldots \otimes \Delta_{k_d}) f \tag{3.16}$$

for $l \in \mathbb{N}$ and multi-index $\mathbf{k} \in \mathbb{N}^d$. In the Smolyak formula we only sum over the simplex $|\mathbf{k}|_1 \leq l + d - 1$, whereas the standard tensor product formula (3.13) results in a summation over the cube $|\mathbf{k}|_\infty \leq l$, because

$$(Q_l^1 \otimes \ldots \otimes Q_l^1) f = \sum_{j=1}^{d} \sum_{1 \leq k_j \leq l} (\Delta_{k_1} \otimes \ldots \otimes \Delta_{k_d}) f.$$

If all nodes in the one-dimensional quadrature formulas are in the order of $\mathcal{O}(2^l)$, the number of nodes in the Smolyak quadrature is of order $n_l^d = \mathcal{O}(2^l l^{d-1})$. This number is significantly smaller than the number of nodes $n_l^d = \mathcal{O}(2^{ld})$ of the tensor product quadrature formula. The nodes of Smolyak's quadrature formula form a *sparse grid* in contrast to the nodes of the full tensor product grid. See Figure 3.5 for a comparison of different sparse grids with full grids.

In order to formulate error bounds for the sparse grid quadrature formula, we first consider the error for one-dimensional quadrature formulas $Q_l^1(f)$ for functions $f \in$

$\mathcal{C}^r(\Omega)$,

$$E_l^1(f) := \left| \int_\Omega f(\mathbf{x}) \, \mathrm{d}\mathbf{x} - Q_l^1(f) \right|. \\ = \mathcal{O}((n_l^1)^{-r}).$$

This bound holds for all quadrature formulas with positive weights, in particular for the Clenshaw-Curtis quadrature formulas. Using such a one-dimensional formula as basis, if $f \in \mathcal{W}_d^r$ and $n_l^1 = \mathcal{O}(2^l)$, the resulting error of the Smolyak formula is given by

$$E_l^d(f) = \mathcal{O}(2^{-lr} l^{(d-1)(r+1)}),$$

whereas the error of the full grid has an order of $\mathcal{O}(2^{-lr/d})$.

Analogously to sparse grid quadrature formulas, also sparse grid interpolation formulas can be defined, see subsection 3.4.3. Sparse grids are widespread in both settings. In the context of uncertainty quantification sparse grids are often used for elliptic partial differential equations with random diffusion coefficients [BTNT12, ES14, FP16]. Besides regular sparse grids, there are also adaptive sparse grids which are either dimension adaptive or spatially adaptive. In dimension-adaptive sparse grids [GG03, Gar07, Gar12] we do not sum over all indices $|\mathbf{k}| \leq l + d - 1$ but only over the admissible index sets I with

$$\mathbf{k} \in I \Rightarrow \mathbf{k} - \mathbf{e}_j \in I, \quad j = 1, \ldots, d, \; k_j > 1,$$

where $\mathbf{e}_j$ denotes the j-th unit vector. Dimension-adaptive sparse grid methods try to find important dimensions and adaptively refine these. Hence, all grid points corresponding to one multi-index $\mathbf{k}$ are refined in one refinement step. In contrast, in spatially-adaptive sparse grid methods only one grid point is refined, see [Pfl10, Pfl12, GK14]. To keep a grid consistent, all missing parents of new grid points have to be created recursively. This spatially adaptive method allows you to place more points near singularities or discontinuities [JAX11].

The Monte Carlo Method

As seen above, tensor product formulas suffer from the curse of dimensionality. The require many integrand evaluations that are exponential in d and, hence, also many simulation runs in the uncertainty quantifications application. Sparse grids overcome this problem only up to a certain limit. In contrast, the curse of dimensionality can be completely be avoided by using Monte Carlo (MC) integration. Monte Carlo methods are based on the Law of Large Numbers.

Theorem 3.23 (The Law of Large Numbers)**.** *If $X_1, X_2, \ldots$ is a sequence of identically distributed and pairwise independent random variables with $\mathbb{E}[|X_i|] < \infty$ for all $i \in \mathbb{N}$, then the sequence satisfies*

1. *the weak law of large numbers*

$$\lim_{n\to\infty} \mathbb{P}\left[\left|\frac{1}{n}\sum_{i=1}^{n} X_i - \mathbb{E}[X_i]\right| > \varepsilon\right] = 0 \quad \forall \varepsilon > 0,$$

   i.e. the sample mean converges in probability to the real mean, and

2. *the strong law of large numbers*

$$\mathbb{P}\left[\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} X_i = \mathbb{E}[X_i]\right] = 1,$$

   i.e. the sample mean converges $\mathbb{P}$-almost sure to the real mean.

There are several variants of the weak law of large numbers. The first variant [Che67] was shown for independent and identically distributed random variables with a finite variance. The next step was to show that the assumption of finite variance is not necessary, see [Khi29]. Later it was proven that it is enough to assume pairwise independent random variables instead of mutually independent ones. The progress of the strong law of large numbers was equivalent. First, it was shown that the strong law is fulfilled for independent and identically distributed random variables with a finite variance and $\sum_{i=1}^{\infty} \mathrm{Var}[X_i]/i^2 < \infty$, see [Kol30]. Then it was shown that the law is true for independent and identically distributed random variables with a finite expectation. The last step was to prove that an assumption of pairwise independent random variables with finite expectation is enough, see [Ete81].

Vanilla Monte Carlo. Assume that one can generate independent and identically distributed samples X_i from the probability measure ρ. Then we can integrate a function f with respect to ρ by applying the law of large numbers to the random variable $Y = f(X)$:

$$\mathbb{E}[f(X)] \approx S_n(f) := \frac{1}{n}\sum_{i=1}^{n} f(X_i).$$

Applying Chebyshev's inequality (3.3) to $S_n(f)$ with the mean $\mu = \mathbb{E}[S_n(f)] = \mathbb{E}[f(X)]$ and the variance

$$\mathrm{Var}[S_n(f)] = \frac{1}{n^2}\sum_{i=1}^{n} \mathrm{Var}[f(X_i)] = \frac{\mathrm{Var}[f(X)]}{n}$$

yields the error estimator

$$\mathbb{P}\left[\left|S_n(f) - \mu\right| \geq k\ \mathrm{Var}[S_n(f)]^{1/2}\right] \leq \frac{1}{k^2}$$

for the Monte Carlo quadrature rule. Choosing $k = \varepsilon^{-1/2}$ we obtain

$$\mathbb{P}\left[\left|S_n(f) - \mu\right| \geq \left(\frac{\mathrm{Var}[f(X)]}{n\varepsilon}\right)^{1/2}\right] \leq \varepsilon.$$

Thus, for a fixed integrand f the error decays with a rate of an order of $\mathcal{O}(n^{-1/2})$ independent from the dimension d of the integration domain or the smoothness of the function. This is the major advantage of Monte Carlo quadrature over tensor product quadrature with a rate of $\mathcal{O}(n^{-r/d})$. But the slow convergence rate of Vanilla Monte Carlo methods is nevertheless a drawback because in many applications it is undesirable to quadruple the number of samples to double the accuracy.

Multilevel Monte Carlo. In many applications the integrand f is associated with the solution of some ordinary differential equation, partial differential equation, or differential algebraic equation, and one can choose the accuracy of the numerical solution. The more accurate a solution is the more expensive it is. This fact is exploited by multilevel Monte Carlo (MLMC) methods. Suppose we have a sequence $f_0, f_1, \ldots, f_L = f$ which approximates f, indexed by a level parameter $0 \leq l \leq L$, with increasing accuracy but also increasing cost, see [Gil15]. Because of the linearity of the expectation, we have the identity

$$\mathbb{E}[f] = \mathbb{E}[f_L] = \mathbb{E}[f_0] + \sum_{l=1}^{L} \mathbb{E}[f_l - f_{l-1}].$$

Each of the summands can be independently estimated using Monte Carlo integration. Thus we obtain the following unbiased estimator for $\mathbb{E}[f]$:

$$\mathbb{E}[f] \approx \frac{1}{n_0} \sum_{i=1}^{n_0} f_0(X_{l,i}) + \sum_{l=1}^{L} \frac{1}{n_l} \sum_{i=1}^{n_l} \left(f_l(X_{l,i}) - f_{l-1}(X_{l,i})\right)$$

where the l in the subscript indicates that independent samples are used at each level.

Let C_0 and V_0 denote the cost and variance of one sample of f_0, and C_l and V_l the cost and variance of one sample of $f_l - f_{l-1}$, then the overall cost C and variance V of the multilevel estimator is given by

$$C = \sum_{l=0}^{L} n_l\, C_l \quad \text{and} \quad V = \sum_{l=0}^{L} \frac{1}{n_l}\, V_l,$$

respectively. To minimize the cost for a fixed variance we use the method of Lagrange multipliers and solve the equation

$$\frac{\partial}{\partial n_l} \sum_{k=0}^{L} \left(n_k C_k + \lambda^2 n_k^{-1} V_k\right) = 0$$

for some Lagrange multiplier λ^2. Thereby, we obtain the minimal cost with $n_l = \lambda\sqrt{V_l/C_l}$. To achieve an overall variance of ε^2 we choose $\lambda = \varepsilon^{-2} \sum_{l=0}^{L} \sqrt{V_l\, C_l}$

resulting in a total cost of

$$C = \varepsilon^{-2} \left(\sum_{l=0}^{L} \sqrt{V_l \, C_l} \right)^2 .$$

It is important to note whether the product $V_l \, C_l$ increases or decreases with increasing l, i.e. whether or not the cost increases with the level faster than the variance decreases. If the product increases with the level, so that the dominant contribution to the cost comes from $V_L \, C_L$ then we have $C \approx \varepsilon^{-2} \, V_L \, C_L$, whereas if it decreases and the dominant contribution comes from $V_0 \, C_0$, then $C \approx \varepsilon^{-2} \, V_0 \, C_0$. In contrast, the standard Monte Carlo cost is approximately $\varepsilon^{-2} V_0 \, C_L$ under the assumption that the cost of computing f_L is similar to the cost of computing $f_L - f_{L-1}$, and that $\mathrm{Var}[f_L] \approx \mathrm{Var}[f_0]$. This shows that in the first case the multilevel Monte Carlo cost is reduced by factor V_L/V_0, whereas in the second case it is reduced by factor C_0/C_L. If the product $V_l \, C_l$ does not vary with level, the total cost is $\varepsilon^{-2} L^2 \, V_0 \, C_0 = \varepsilon^{-2} L^2 \, V_L \, C_L$.

The Quasi-Monte Carlo Method

In this section we introduce the method of quasi-Monte Carlo sampling. Analogously to Monte Carlo integration, the quasi-Monte Carlo quadrature rule has weights $w_i = \frac{1}{n}$, i.e. the integral of a function f is approximated by the average of the function evaluated at a set of points $P = \{x_1, \ldots, x_n\}$

$$\int_{[0,1]^d} f(x) \, \mathrm{d}x \approx \frac{1}{n} \sum_{i=1}^{n} f(x_i).$$

The difference between quasi-Monte Carlo and Monte Carlo quadrature is the way the sampling points x_i are chosen. Quasi-Monte Carlo methods use a low-discrepancy sequence such as the Halton sequence [Hal60, Nie92, Owe06] or the Sobol sequence [Sob58, Nie92]. The points are deterministically sampled but in such a way that they appear to be quite random. The advantage over Monte Carlo sampling is that the sample space is covered more uniformly which results in a faster rate of convergence. Quasi-Monte Carlo quadrature has a rate of convergence close to $\mathcal{O}(n^{-1})$, whereas the rate for the Monte Carlo method is $\mathcal{O}(n^{-1/2})$.

A way to measure how uniformly distributed the sampling points are, is the star discrepancy $D^*(P)$, which examines the difference between the real volume of some set B and its sampled volume.

Definition 3.24 (Discrepancy)**.** The star discrepancy $D^*(P)$ for a set of points $P = \{x_1, \ldots, x_n\} \subseteq [0,1]^d$ is defined as

$$D^*(P) = \sup_{B \subset [0,1]^d} \left| \frac{\#(P \cap B)}{n} - \mathrm{vol}(B) \right|,$$

where $B = \prod_{i=1}^{d} [0, b_i)$ is a rectangular solid in $[0,1]^d$ with sides that are parallel to the coordinate axes.

The error of quasi-Monte Carlo quadrature is typically computed for functions of bounded Hardy-Krause variation $V^{\mathrm{HK}}(f)$. As in [Gla13] consider a rectangle of the form

$$J = [u_1^-, u_1^+] \times \ldots \times [u_d^-, u_d^+]$$

with $0 \leq u_i^- \leq u_i^+ \leq 1$ for $i = 1, \ldots, d$. Each vertex u of J has coordinates of the form $u_i^{\pm}$. We divide the set of vertices into a set of vertices of J with an even number of $+$ superscripts, $\mathcal{E}(J)$, and a set of vertices of J with an odd number of $+$ superscripts, $\mathcal{O}(J)$, and define:

$$\Delta(f, J) = \sum_{u \in \mathcal{E}(J)} f(u) - \sum_{u \in \mathcal{O}(J)} f(u).$$

This is the sum of the values of f at the 2^d vertices of J with alternating signs at nearest-neighbor vertices.

Definition 3.25 (Vitali Variation)**.** Let $\mathcal{P}$ be a partition of $[0,1]^d$ into finitely many non-overlapping rectangles of the form J, then

$$V^{(d)}(f) = \sup_{\mathcal{P}} \sum_{J \in \mathcal{P}} |\Delta(f, J)|$$

is called Vitali variation of the function f.

Summing up the Vitali variation over all faces of $[0,1]^d$ yields the Hardy-Krause variation.

Definition 3.26 (Hardy-Krause Variation)**.** For any $1 \leq k \leq d$ and any $1 \leq i_1 < i_2 < \ldots < i_k \leq d$, consider the function on $[0,1]^k$ defined by restricting f to points $(u_1, \ldots, u_d)$ with $u_j = 1$ if $j \notin I^{(k)} = \{i_1, \ldots, i_k\}$ and $(u_{i_1}, \ldots, u_{i_k})$ ranging over all of $[0,1]^k$. Denote by $V^{(k)}(f, I^{(k)})$ the application of $V^{(k)}$ to this function. Then the Hardy-Krause variation of f is defined as

$$V(f) = \sum_{k=1}^{d} \sum_{I^{(k)}} V^{(k)}(f, I^{(k)}).$$

Now we have all parts to estimate the error ε of the quasi-Monte Carlo quadrature rule

$$\varepsilon = \left| \int_{[0,1]^d} f(x) \, \mathrm{d}x - \frac{1}{n} \sum_{i=1}^{n} f(x_i) \right|$$

with quasi-Monte Carlo sampling points $x_1, \ldots, x_n$. The Koksma-Hlawka inequality states that the error is bounded by a term proportional to the discrepancy of the set $P = \{x_1, \ldots, x_n\}$.

Theorem 3.27. *Let $f : [0,1]^d \to \mathbb{R}$ have a bounded Hardy-Krause variation V(f), then for any set $P = \{x_1, \dots, x_n\} \subseteq [0,1)^d$,*

$$\left| \frac{1}{n} \sum_{i=1}^{n} f(x_i) - \int_{[0,1]^d} f(x) \, \mathrm{d}x \right| \leq V(f) \, D^*(P).$$

The Koksma–Hlawka inequality is sharp in the following sense: for every $P = \{x_1, \dots, x_n\} \subseteq [0,1)^d$ and every $\varepsilon > 0$ there is a function f with bounded variation and $V(f) = 1$ such that

$$\left| \frac{1}{n} \sum_{i=1}^{n} f(x_i) - \int_{[0,1]^d} f(x) \, \mathrm{d}x \right| > D^*(P) - \varepsilon.$$

Therefore, the quality of a numerical integration rule depends only on the discrepancy $D^*(P)$.

For a proof see [Nie92].

The inequality $|\varepsilon| \leq V(f) \, D^*(P)$ can be used to show that the integration error of the quasi-Monte Carlo method is $\mathcal{O}(n^{-1}(\log n)^d)$, see [Nie92]. Though we can only state the upper bound of the approximation error, the convergence rate of the quasi-Monte Carlo method is in practice usually much faster than its theoretical bound. Hence, in general, the accuracy of the quasi-Monte Carlo method increases faster than that of the Monte Carlo method. However, this advantage is only guaranteed if n is large enough and the variation is finite.

The Halton Sequence. An often used low discrepancy sequence is the Halton sequence [Hal60, Nie92], which uses co-prime numbers $b_1, \dots, b_d$ as its bases. First consider the one-dimensional case. For this let

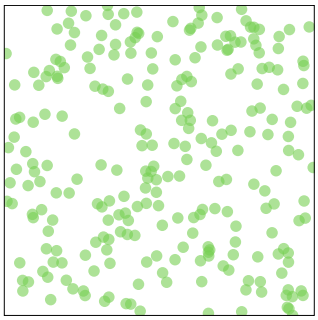
$$n = \sum_{k=0}^{K} d_k(n) b^k$$

be the b-ary representation of the positive integer $n \geq 1$, i.e. $0 \leq a_k(n) < b$. Then inversion and shifting yields the n-th Halton point

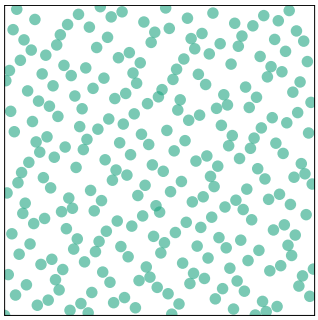
$$x_n^{(b)} = \sum_{k=0}^{K} d_k(n) b^{-k-1}.$$

Pairing d sequences for co-prime bases $b_1, \dots, b_d > 1$ yields the d-dimensional Halton sequence

$$x_n = \left(x_n^{(b_1)}, \dots, x_n^{(b_d)} \right)$$



(a) Monte Carlo points.



(b) Halton points.

Figure 3.6: Comparison of 256 Monte Carlo (left) and quasi-Monte Carlo (right) points in two dimensions. The gaps between the Monte Carlo points are significantly larger than between the quasi-Monte Carlo points.

with star discrepancy $D^*(P) \leq C\frac{(\log n)^d}{n}$ where the constant C only depends on $b_1, \ldots, b_d$.

Example 3.28. Consider the Halton sequence with basis 2 dividing the interval $(0, 1)$ in 2^k sub-intervals

$$\tfrac{1}{2},\ \tfrac{1}{4},\ \tfrac{3}{4},\ \tfrac{1}{8},\ \tfrac{5}{8},\ \tfrac{3}{8},\ \tfrac{7}{8},\ \ldots$$

and the Halton sequence with basis 3 dividing the interval in 3^k sub-intervals

$$\tfrac{1}{3},\ \tfrac{2}{3},\ \tfrac{1}{9},\ \tfrac{4}{9},\ \tfrac{7}{9},\ \tfrac{8}{9},\ \tfrac{1}{27},\ \ldots.$$

The resulting two-dimensional sequence is given by

$$\left(\tfrac{1}{2}, \tfrac{1}{3}\right),\ \left(\tfrac{1}{4}, \tfrac{2}{3}\right),\ \left(\tfrac{3}{4}, \tfrac{1}{9}\right),\ \left(\tfrac{1}{8}, \tfrac{4}{9}\right),\ \left(\tfrac{5}{8}, \tfrac{7}{9}\right),\ \left(\tfrac{3}{8}, \tfrac{2}{9}\right),\ \left(\tfrac{7}{8}, \tfrac{5}{9}\right),\ \left(\tfrac{1}{16}, \tfrac{8}{9}\right),\ \left(\tfrac{9}{16}, \tfrac{1}{27}\right), \ldots.$$

See Figure 3.6 for the difference between Monte Carlo points and these Halton points.

Due to its simplicity and weak assumptions, all types of Monte Carlo integration are used in various research areas, e.g. in meteorology [TGG13, AFM06, KBJ14], hydro-geology [CGST11, CGP17, SST17], medicine [Pag12, SBS00, KKSV00], biology [KBK+08, JC12, KK05], or finance [KKK10, Ale01, Gla13].

Summary

All previously discussed quadrature methods have different advantages and disadvantages. The stronger the assumptions of the method are, the higher is the possible convergence rate. The full grid quadrature method has a convergence rate of $-r/d$ for functions with bounded derivatives up to order r but suffers from the curse of dimensionality. To achieve a given accuracy, we need an exponential in d number of quadrature points. Thinning out the full grid of n^d points to a sparse grid of $n(\log n)^{d-1}$ points overcomes this curse of dimensionality up to a certain extent. Instead of bounded derivatives, sparse grid quadrature methods need the assumption

Method	Function Space	Error
Full Grid	$\mathcal{C}^r$	$\mathcal{O}\left(n^{-r/d}\right)$
Sparse Grid	$\mathcal{W}^r$	$\mathcal{O}\left(n^{-r}(\log n)^{(d-1)(r+1)}\right)$
Monte Carlo	bounded expectation	$\mathcal{O}\left(n^{-1/2}\right)$
Quasi-Monte Carlo	bounded variation	$\mathcal{O}\left(n^{-1}(\log n)^{d}\right)$

Table 3.1: Comparison of required function spaces and error bounds for different quadrature formulas with the dimension d, the number n of quadrature points, and the regularity r.

of bounded mixed derivatives of order r. Using the higher regularity of the integrand leads to a higher convergence rate. Both methods, full grid and sparse grid quadrature, have the disadvantage that they require smooth integrands. In contrast, Monte Carlo methods are universally usable because they do not need a smooth integrand but only an integrand with bounded expectation. Their convergence rate of $-1/2$ is independent from the dimension d but extremely low. To double the accuracy, the number of quadrature points must be quadrupled. Choosing the quadrature points more structuredly and less randomly results in an improved convergence rate for quasi-Monte Carlo methods. Table 3.1 provides a short overview of the methods including the requirements and the error bounds.

3.3 Spectral Expansions

The idea of spectral expansions is to represent a random quantity with an expansion consisting of functions of random variables multiplied with deterministic coefficients. In the first two parts of this section we introduce two different types of spectral expansions. The Karhunen-Loève expansion represents the random field in terms of its spatial correlation and can be used to model epistemic uncertainty, e.g. the permeability of porous media [JEX10]. In contrast, the polynomial chaos expansion represents the random field in terms of its stochastic dimension and is ideal to describe aleatory uncertainties such as noisy signals. The Karhunen-Loève expansion is applicable to correlated processes and decorrelates them, whereas the polynomial chaos expansion needs independent random variables with tensor product structure. Therefore, the polynomial chaos expansion separates random dimensions from deterministic dimensions, and the contributions from each uncertain input can be easily identified. In the last two subsections, we discuss two different possibilities, namely the intrusive methods and the non-intrusive methods, to compute the coefficients of a spectral expansion. Intrusive methods reformulate the original model into a stochastic version and search for a solution of the new model, whereas non-intrusive methods incorporate several solutions of the original problem. The Galerkin projection, an intrusive method, is described in the third subsection, and the non-intrusive spectral projection is discussed in the last subsection.

For the formal definition of a spectral expansion let $(\Omega, \mathcal{A}, \mathbb{P})$ be a probability space with an event space Ω and a probability measure $\mathbb{P}$ defined on the σ-algebra $\mathcal{A}$ of subsets of Ω. Let $\xi(\omega)$ be a random variable on Ω. We consider as in [PNI15] second-order random fields or square integrable random fields, i.e. functions u belonging to the space

$$L^2(\Omega, \mathbb{P}) = \left\{ f \text{ measurable w.r.t. } \mathbb{P}, \int_\Omega f^2(\xi)\, \mathrm{d}\mathbb{P}(\xi) < \infty \right\}.$$

The inner product of two functionals $f, g \in L^2(\Omega, \mathbb{P})$ is defined by

$$\langle f, g \rangle = \int_\Omega f(\xi) g(\xi)\, \mathrm{d}\mathbb{P}(\xi).$$

This inner product induces the norm $\|f\|^2 = \langle f, f \rangle$. A spectral expansion of a random functional is of the form

$$f(\xi) = \sum_{k=0}^{\infty} f_k \psi_k(\xi),$$

where $\{\psi_k(\xi)\}_{k=0}^{\infty}$ is the set of basis functions and f_k the set of coefficients to be determined by $f_k = \langle f, \psi_k \rangle$.

3.3.1 The Karhunen-Loève Expansion

Let $\mathcal{X} \subset \mathbb{R}^d$ be a bounded domain, which could be a spatial, temporal, or general parameter space with a probability space $(\Omega, \mathcal{A}, \mathbb{P})$. Consider a zero-mean second order random field $u : \mathcal{X} \times \Omega \to \mathbb{R}$ with continuous covariance function $C_u(\mathbf{x}, \mathbf{y}) = \mathbb{E}[u(\mathbf{x})u(\mathbf{y})]$. We associate to C_u a linear operator $T_{C_u} : L^2(\mathcal{X}) \to L^2(\mathcal{X})$ defined by

$$(T_{C_u} u)(\mathbf{x}) = \int_{\mathcal{X}} C_u(\mathbf{x}, \mathbf{y}) u(\mathbf{y})\, \mathrm{d}y.$$

The covariance function C_u is a *Mercer kernel*, i.e. C_u is continuous, symmetric, and positive semi-definite: for all choices of finitely many points $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathcal{X}$, the Gram matrix

$$G := \begin{bmatrix} C_u(\mathbf{x}_1, \mathbf{x}_1) & \ldots & C_u(\mathbf{x}_1, \mathbf{x}_n) \\ \vdots & \ddots & \vdots \\ C_u(\mathbf{x}_n, \mathbf{x}_1) & \ldots & C_u(\mathbf{x}_n, \mathbf{x}_n) \end{bmatrix}$$

is positive semi-definite. Hence, by Mercer's theorem [Mer09], the eigenfunctions ϕ_k of T_{C_u} with corresponding eigenvalues $\lambda_k > 0$,

$$\int_{\mathcal{X}} C_u(\mathbf{x}, \mathbf{y}) \phi_k(\mathbf{y})\, \mathrm{d}\mathbf{y} = \lambda_k \phi_k(\mathbf{x}), \quad k = 1, 2, \ldots, \tag{3.17}$$

form an orthonormal basis of $L^2(\mathcal{X})$. Furthermore, the covariance function has the spectral decomposition

$$C_u(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{\infty} \lambda_k \phi_k(\mathbf{x}) \phi_k(\mathbf{y}),$$

which converges absolutely and uniformly over compact subsets of $\mathcal{X}$.

The Karhunen-Loève expansion of the random field u provides a series representation in terms of its spatial correlation and is bi-orthogonal since the random coefficients are orthogonal in the probability space while the deterministic functions are orthogonal in the spatial domain.

Theorem 3.29 (Karhunen-Loève)**.** *Let $\mathcal{X}$ be bounded and $u : \mathcal{X} \times \Omega \to \mathbb{R}$ a zero-mean second order random field with a continuous and square-integrable covariance function. Then $C_u(\mathbf{x}, \mathbf{y})$ is a Mercer kernel and the eigenfunctions ϕ_k with decreasing eigenvalues $\lambda_k > 0$ form an orthonormal basis of $L^2(\mathcal{X})$. The Karhunen-Loève expansion of u is defined as*

$$u(\mathbf{x}) = \sum_{k=1}^{\infty} Z_k \phi_k(\mathbf{x}) \tag{3.18}$$

where the convergence is in $L^2(\Omega)$ and

$$Z_k = \int_{\mathcal{X}} u(\mathbf{x}) \phi_k(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$

The random variables Z_k are centered, uncorrelated, and have the variance λ_k, i.e. $\mathbb{E}[Z_k] = 0$ and $\mathbb{E}[Z_k Z_l] = \lambda_k \delta_{kl}$.

Proof. The representation as linear combination of orthonormal basis functions follows directly by Mercer's theorem. For the random variables Z_k it holds that

$$\mathbb{E}[Z_k] = \mathbb{E}\left[\int_{\mathcal{X}} u(\mathbf{x}) \phi_k(\mathbf{x}) \, \mathrm{d}\mathbf{x}\right] = \int_{\mathcal{X}} \mathbb{E}[u(\mathbf{x})] \phi_k(\mathbf{x}) \, \mathrm{d}\mathbf{x} = 0$$

and

$$\begin{aligned}
\mathbb{E}[Z_k Z_l] &= \mathbb{E}\left[\int_{\mathcal{X}} \int_{\mathcal{X}} u(\mathbf{x}) u(\mathbf{y}) \phi_k(\mathbf{x}) \phi_l(\mathbf{y}) \, \mathrm{d}\mathbf{x} \, \mathrm{d}\mathbf{y}\right] \\
&= \int_{\mathcal{X}} \int_{\mathcal{X}} \mathbb{E}\left[u(\mathbf{x}) u(\mathbf{y})\right] \phi_k(\mathbf{x}) \phi_l(\mathbf{y}) \, \mathrm{d}\mathbf{x} \, \mathrm{d}\mathbf{y} \\
&= \int_{\mathcal{X}} \int_{\mathcal{X}} C_u(\mathbf{x}, \mathbf{y}) \phi_k(\mathbf{x}) \phi_l(\mathbf{y}) \, \mathrm{d}\mathbf{x} \, \mathrm{d}\mathbf{y} \\
&= \int_{\mathcal{X}} \phi_l(\mathbf{y}) \left(\int_{\mathcal{X}} C_u(\mathbf{x}, \mathbf{y}) \phi_k(\mathbf{x}) \, \mathrm{d}\mathbf{x}\right) \mathrm{d}\mathbf{y} \\
&= \lambda_k \int_{\mathcal{X}} \phi_k(\mathbf{x}) \phi_l(\mathbf{y}) \, \mathrm{d}\mathbf{y} \\
&= \lambda_k \, \delta_{kl}.
\end{aligned}$$

Hence, the variables are uncorrelated and have the variance λ_k. To see that the expansion (3.18) converges in $L^2(\Omega)$, let

$$u_N(\mathbf{x}) = \sum_{k=1}^{N} Z_k \phi_k(\mathbf{x})$$

be the truncated Karhunen-Loève expansion. Then

$$\begin{aligned}
&\mathbb{E}[(u(\mathbf{x}) - u_N(\mathbf{x}))^2] \\
&\quad = \mathbb{E}[(u(\mathbf{x}))^2] - 2\mathbb{E}[u(\mathbf{x}) f_N(\mathbf{x})] + \mathbb{E}[(u_N(\mathbf{x}))^2] \\
&\quad = C_u(\mathbf{x},\mathbf{x}) - 2\mathbb{E}\left[u(\mathbf{x}) \sum_{k=1}^{N} Z_k \phi_k(\mathbf{x})\right] + \mathbb{E}\left[\sum_{k=1}^{N}\sum_{l=1}^{N} Z_k Z_l \phi_k(\mathbf{x})\phi_l(\mathbf{x})\right] \\
&\quad = C_u(\mathbf{x},\mathbf{x}) - 2\mathbb{E}\left[\sum_{k=1}^{N} \int_{\mathcal{X}} u(\mathbf{x})u(\mathbf{y})\phi_k(\mathbf{x})\phi_k(\mathbf{y})\,\mathrm{d}\mathbf{y}\right] + \sum_{k=1}^{N} \lambda_k \phi_k(\mathbf{x})^2 \\
&\quad = C_u(\mathbf{x},\mathbf{x}) - \sum_{k=1}^{N} \lambda_k \phi_k(\mathbf{x})^2
\end{aligned}$$

which tends to 0 by Mercer's theorem. □

The truncated Karhunen-Loève expansion is optimal in the mean square sense, i.e. the total mean square error

$$\int_{\mathcal{X}} \mathbb{E}[(u - u_N)^2]\,\mathrm{d}\mathbf{x} = \int_{\mathcal{X}} \sum_{k>N} \lambda_k \phi_k^2(\mathbf{x})\,\mathrm{d}\mathbf{x} = \sum_{k>N} \lambda_k$$

is minimized if the ψ_k are chosen to be the eigenfunctions of T_{C_u}. No other approximation of u in a series of N terms results in a smaller error. The total mean square error decreases monotonically with N at a rate that depends on the decay of the spectrum of C_u. The higher the rate of the spectral decay is, the smaller is the number of terms needed in the expansion. Specifically, the rate depends on the correlation function of the process, see [MK10]. The more correlated the process is, the higher is the rate and hence the smaller is the number of terms needed to achieve a desired threshold. In the limit when $C_u(x,y) = 1$, which implies an infinite correlation length and that the process $u(x)$ is fully correlated, the process depends on just one random variable. If the process is poorly correlated, a higher number of terms is needed. Moreover, in the limit of diminishing correlation length, where u corresponds to white noise, i.e. $C_u(x,y) \sim \delta(x-y)$, any set of orthogonal functions can be the eigenfunctions and the eigenvalues are constant, i.e. $\lambda_k = 1$. Hence, an infinite number of terms would be necessary to achieve a given threshold.
As an example, consider the exponential covariance function $C_u(x,y) = \exp(-|x - y|/a)$ where $a > 0$ is the correlation length and let $x, y \in [-1, 1]$. Then the eigenvalue problem (3.17) can be solved analytically and the eigenvalues are given by

$$\lambda_k = \begin{cases} \frac{2a}{1+a^2 w_k^2}, & k \text{ even} \\ \frac{2a}{1+a^2 v_k^2}, & k \text{ odd} \end{cases}$$

where w_k and v_k are the solutions of the equations

$$\begin{cases} aw_k + \tan(w_k) = 0, & k \text{ even} \\ 1 - av_k \tan(v_k) = 0, & k \text{ odd.} \end{cases}$$

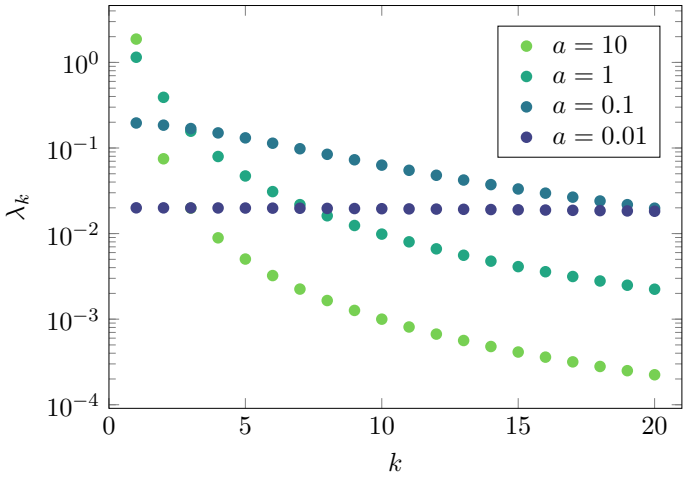


Figure 3.7: The first 20 eigenvalues of the exponential covariance function with different correlation lengths a.

The eigenvalues are shown in Figure 3.7 for several different correlation lengths a. It can be seen that the eigenvalues decay and that the decay rate is larger when the correlation length is longer. When the correlation length is very small, e.g. $a = 0.01$, the decay of the eigenvalues is hardly visible.

Since the random variables Z_k of the Karhunen-Loève expansion are uncorrelated, the variance of the random field can be easily obtained by

$$\mathrm{Var}[u(\mathbf{x})] = \sum_{k=1}^{\infty} \mathrm{Var}[Z_k]\phi_k(\mathbf{x})^2 = \sum_{k=1}^{\infty} \lambda_k \phi_k(\mathbf{x}).$$

Integration over the spatial domain $\mathcal{X}$ and using the orthonormality of the basis functions ϕ_k yields the total variance

$$\int_{\mathcal{X}} \mathrm{Var}[u(\mathbf{x})]\, \mathrm{d}\mathbf{x} = \sum_{k=1}^{\infty} \lambda_k.$$

Hence, the truncated Karhunen-Loève expansion captures $\sum_{k=1}^{N} \lambda_k / \sum_{k=1}^{\infty} \lambda_k$ of the total variance.

Typical applications of the Karhunen-Loève expansion are groundwater flow problems in porous media [ES14, CQ15, CGP17, BD17, LZ07, TMEP11]. Since it is physically impossible to know the exact permeability at every point in the domain, the permeability is modeled as a random field with an experimentally determined covariance structure, see [GKW+07]. To determine the covariance function, a lot of physical measurements and data fitting is necessary, see [DB12, Des87]. The diffusion coefficient a is modeled as a two-dimensional log-normal Gaussian random field to ensure positive permeability almost surely, i.e.

$$a(\mathbf{x}, \omega) = \exp\left(\phi_0 + \sum_{k=1}^{\infty} \sqrt{\lambda_k}\psi_k(\mathbf{x})\xi_k(\omega)\right)$$

where $\psi_0 = \mathbb{E}[\log a(\mathbf{x}, \cdot)]$ and ξ_k are mutually uncorrelated Gaussian random variables with a mean of zero and a unit variance. In this application the covariance kernel is given by

$$C_{\log a}(\mathbf{x}, \mathbf{y}) = \sigma^2 \exp\left(-\frac{|x_1 - y_1|}{\eta_1} - \frac{|x_2 - y_2|}{\eta_2}\right),$$

where σ and η_i denote the variance and correlation length in the i-th spatial dimension, respectively. In this case, the eigenfunction and eigenvalues can be computed analytically, see [ZL04].

The Discrete Karhunen-Loève Expansion. Now consider a discrete and finite set $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ instead of the domain $\mathcal{X}$. Then $U = (u(\mathbf{x}_1), \ldots, u(\mathbf{x}_n))^T$ is an n-dimensional random vector with the covariance matrix

$$\Sigma_{ij} = \mathbb{E}[u(\mathbf{x}_i)u(\mathbf{x}_j)], \quad \forall i, j \in \{1, \ldots, n\}.$$

Formulating the integral equation (3.17) in this discrete case yields the matrix eigenvalue problem

$$\boldsymbol{\Sigma}\boldsymbol{\phi}_k = \lambda_k \boldsymbol{\phi}_k$$

where $\boldsymbol{\phi}_k = (\phi_{k,1}, \ldots, \phi_{k,n})^T$ is an n-dimensional vector. Since $\boldsymbol{\Sigma}$ is positive definite and symmetric, its eigenvectors are orthonormal and form a basis of $\mathbb{R}^n$. Let $(\lambda_k, \boldsymbol{\phi}_k)$ be the resulting eigenpairs listed in decreasing order of λ_k, and let $\boldsymbol{\Phi} = (\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_n)^T$ be the orthonormal matrix of eigenvectors, then the discrete Karhunen-Loève transformation [DG13] of u reads as

$$\begin{aligned} U &= \sum_{k=1}^{n} \langle \boldsymbol{\phi}_k, U \rangle \boldsymbol{\phi}_k \\ &= \sum_{k=1}^{n} Z_k \boldsymbol{\phi}_k \end{aligned}$$

or in matrix form as

$$\begin{aligned} Z &= \boldsymbol{\Phi}^T U \\ U &= \boldsymbol{\Phi} Z. \end{aligned}$$

Hence, the discrete Karhunen-Loève expansion of u results in the well-known principal component transformation of U.

3.3.2 Polynomial Chaos Expansions

The Wiener-Hermite Polynomial Chaos

In contrast to the Karhunen-Loève expansion, the polynomial chaos expansion is a series representation of a random field in terms of its stochastic dimension and not its spatial dimension. First, we will consider the original polynomial chaos (PC) expansion with Hermite polynomials, which has been introduced by Wiener [Wie38].

The idea of the original polynomial chaos expansion is to find an expansion of a real-valued random variable $u(\xi)$ with respect to a standard Gaussian random variable ξ in the orthogonal basis of Hermite polynomials. The Hermite polynomials are defined as

$$H_n(\xi) = (-1)^n e^{\xi^2/2} \frac{\mathrm{d}^n}{\mathrm{d}\xi^n} e^{-\xi^2/2}, \quad n = 0, 1, \ldots.$$

They fulfill the recursive relation

$$\begin{aligned} H_0(\xi) &= 1, \\ H_1(\xi) &= \xi, \\ H_{n+1}(\xi) &= \xi H_n(\xi) - n H_{n-1}(\xi), \end{aligned}$$

and the differential rule

$$\frac{\mathrm{d}}{\mathrm{d}\xi} H_n(\xi) = n H_{n-1}(\xi). \tag{3.19}$$

The Hermite polynomials are orthogonal with respect to the inner product according to the standard Gaussian measure $\rho(\xi)$, i.e.

$$\langle H_n, H_m \rangle := \frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} H_n(\xi) H_m(\xi)\, e^{-\frac{\xi^2}{2}} \mathrm{d}\xi = n!\, \delta_{n,m}$$

They form a complete basis of the set of square integrable functions with respect to the standard Gaussian measure, the Hilbert space $L^2(\mathbb{R}, \rho)$ with norm $\|u\| = \sqrt{\langle u, u \rangle}$. Hence, for any $u \in L^2(\mathbb{R}, \rho)$ there exists the Wiener-Hermite polynomial chaos expansion

$$u(\xi) = \sum_{n=0}^{\infty} u_n H_n(\xi) \tag{3.20}$$

with coefficients defined by the projection

$$u_n = \frac{\langle u, H_n \rangle}{\langle H_n, H_n \rangle} = \frac{1}{\sqrt{2\pi} n!} \int_{\mathbb{R}} u(\xi) H_n(\xi)\, e^{-\frac{\xi^2}{2}}\, \mathrm{d}\xi.$$

In practice, the truncated Wiener-Hermite polynomial chaos expansion

$$u_N(\xi) = \sum_{n=0}^{N} u_n H_n(\xi) \tag{3.21}$$

is often used, since it converges very fast when the function $u(\xi)$ is very smooth. The truncation error $u - u_N$ is orthogonal to the subspace spanned by the Hermite polynomials of degree at most N and tends to zero in mean square, cf. [CM47].

Lemma 3.30. *The truncation error u_N is orthogonal to the subspace*

$$\operatorname{span}\{H_0, H_1, \ldots, H_N\}$$

of $L^2(\mathbb{R}, \rho)$. Furthermore, $\lim_{N\to\infty} u_N = u$ in $L^2(\mathbb{R}, \rho)$.

Proof. Let $v := \sum_{m=0}^{N} v_m H_m$ be any element of the subspace $\operatorname{span}\{H_0, H_1, \ldots, H_N\}$, then

$$\begin{aligned}\langle u - u_N, v\rangle &= \left\langle \sum_{n=N+1}^{\infty} u_n H_n, \sum_{m=0}^{N} v_m H_m \right\rangle \\ &= \sum_{n=N+1}^{\infty} \sum_{m=0}^{N} u_n\, v_m\, \langle H_n, H_m\rangle \\ &= 0.\end{aligned}$$

Thereby, using Pythagoras' theorem

$$\|u\|^2 = \|u_N\|^2 + \|u - u_N\|^2,$$

and since $\|u - u_N\| \to 0$ as $N \to \infty$ by [CM47]

$$\lim_{N\to\infty} \|u_N\|^2 = \|u\|^2.$$

□

Thus, polynomial chaos provides a means for expanding second-order random processes – i.e. most physical processes – in terms of Hermite polynomials. The following theorem [AGP+08] shows the fast convergence rate of the Wiener-Hermite polynomial chaos expansion for smooth functions.

Theorem 3.31. *Let $u(\xi) \in \mathcal{C}^k(\mathbb{R})$ with the truncated Wiener-Hermite polynomial chaos expansion expansion $u_N(\xi) = \sum_{n=0}^{N} u_n H_n(\xi)$, then the error can be estimated by*

$$\|u - u_N\|^2 \leq \begin{cases} \dfrac{\|u^{(k)}\|^2}{\prod_{l=0}^{k-1}(N+1-l)}, & k \leq N \\ \dfrac{\|u^{(N+1)}\|^2}{(N+1)!}, & k > N. \end{cases}$$

Proof. Recall the differentiation rule (3.19) for Hermite polynomials. Applying it k times yields

$$H_n^{(k)}(\xi) = \prod_{l=0}^{k-1}(n-l) H_{n-k}(\xi).$$

Using this identity we get

$$\begin{aligned}\|u^{(k)}\|^2 &= \left\langle \sum_{n=k}^{\infty} u_n \prod_{l=0}^{k-1}(n-l)H_{n-k},\ \sum_{m=k}^{\infty} u_m \prod_{l=0}^{k-1}(m-l)H_{m-k} \right\rangle \\ &= \sum_{n=k}^{\infty}\sum_{m=k}^{\infty} u_n u_m \prod_{l=0}^{k-1}(n-l) \prod_{l=0}^{k-1}(m-l)\ \langle H_{n-k}, H_{m-k}\rangle \\ &= \sum_{n=k}^{\infty} u_n^2\ (n-k)! \prod_{l=0}^{k-1}(n-l)^2.\end{aligned}$$

First, consider the case $k \leq N$. Then for the approximation of u by the truncated Hermite expansion u_N it holds

$$\begin{aligned}\|u-u_N\|^2 &= \sum_{n=N+1}^{\infty} u_n^2\, n! \\ &= \sum_{n=N+1}^{\infty} u_n^2\ (n-k)! \prod_{l=0}^{k-1}(n-l) \\ &\leq \sum_{n=N+1}^{\infty} u_n^2\ (n-k)! \prod_{l=0}^{k-1}(n-l) \frac{\prod_{l=0}^{k-1}(n-l)}{\prod_{l=0}^{k-1}(N+1-l)} \\ &\leq \frac{\|u^{(k)}\|^2}{\prod_{l=0}^{k-1}(N+1-l)}.\end{aligned}$$

Now consider the case $k > N$

$$\begin{aligned}\|u-u_N\|^2 &= \sum_{n=N+1}^{\infty} u_n^2\, n! \\ &= \sum_{n=N+1}^{\infty} u_n^2\ (n-N+1)! \prod_{l=0}^{N}(n-l) \\ &\leq \sum_{n=N+1}^{\infty} u_n^2\ (n-N+1)! \prod_{l=0}^{N}(n-l) \frac{\prod_{l=0}^{N}(n-l)}{\prod_{l=0}^{N}(N+1-l)} \\ &\leq \frac{\|u^{(N+1)}\|^2}{(N+1)!}.\end{aligned}$$

□

Therefore, the rate of convergence of the truncated Wiener-Hermite polynomial chaos expansion relies on the smoothness of $u(\xi)$. For a fixed number N of terms holds the smoother the function u is, the smaller is the approximation error. In the literature this kind of convergence rate is referred to as spectral convergence. If $u(\xi)$ is infinitely smooth, using the Stirling formula

$$n! \sim \sqrt{2\pi n}\left(\frac{n}{e}\right)^n, \quad \text{for } n \to \infty$$

yields the exponential convergence rate

$$\|u - u_N\|^2 \leq ce^{-qN}.$$

Instead of a random variable $u(\xi)$ we now consider a random field $u : \mathcal{X} \times \Omega \to \mathbb{R}$ with the spatial (and/or temporal) domain $\mathcal{X}$ and the event set $\Omega = \mathbb{R}$ of a Gaussian random variable ξ. Then the Wiener-Hermite polynomial chaos expansion reads

$$u(\mathbf{x}, \xi) = \sum_{n=0}^{\infty} u_n(\mathbf{x}) H_n(\xi)$$

and the deterministic coefficients u_n are functions of the spatial variable $\mathbf{x}$. As stated above this approach separates random dimensions from deterministic ones. Once one has determined the coefficients of the polynomial chaos expansion, either with a Galerkin projection (Section 3.3.3) or a non-intrusive spectral projection (Section 3.3.4), several statics of the random field, such as expectation, variance, or covariance can be easily obtained. Since $H_0 \equiv 1$ we obtain for $\mathbf{x} \in \mathcal{X}$

$$\begin{aligned}
\mathbb{E}[u(\mathbf{x}, \xi)] &= \langle H_0, u \rangle \\
&= \sum_{n=0}^{\infty} u_n(\mathbf{x}) \langle H_0, H_n \rangle \\
&= u_0(\mathbf{x}).
\end{aligned}$$

Since

$$\begin{aligned}
\mathbb{E}[u^2(\mathbf{x}, \xi)] &= \left\langle \sum_{n=0}^{\infty} u_n(\mathbf{x}) H_n, \sum_{m=0}^{\infty} u_m(\mathbf{x}) H_m \right\rangle \\
&= \sum_{n,m=0}^{\infty} u_n(\mathbf{x}) u_m(\mathbf{x}) \langle H_n, H_m \rangle \\
&= \sum_{n=0}^{\infty} u_n^2(\mathbf{x}) n!,
\end{aligned}$$

it follows for the variance

$$\begin{aligned}
\mathrm{Var}[u(\mathbf{x}, \xi)] &= \mathbb{E}[u(\mathbf{x}, \xi)^2] - \mathbb{E}[u(\mathbf{x}, \xi)]^2 \\
&= \sum_{n=1}^{\infty} u_n(\mathbf{x})^2 \, n!.
\end{aligned}$$

In view of the expression for the variance, the polynomial chaos coefficients can be used as sensitivity indices. A natural measure of how strongly u depends upon H_n is given by the fraction

$$\frac{u_n^2 n!}{\sum_{n=1}^{\infty} u_n(\mathbf{x})^2 \, n!}.$$

Analogously, for two points $\mathbf{x}, \mathbf{y} \in \mathcal{X}$

$$\begin{aligned}\mathbb{E}[u(\mathbf{x},\xi)u(\mathbf{y},\xi)] &= \left\langle \sum_{n=0}^{\infty} u_n(\mathbf{x})H_n, \sum_{m=0}^{\infty} u_m(\mathbf{y})H_m \right\rangle \\ &= \sum_{n,m=0}^{\infty} u_n(\mathbf{x})u_m(\mathbf{y})\langle H_n, H_m\rangle \\ &= \sum_{n=0}^{\infty} u_n(\mathbf{x})u_n(\mathbf{y})n!,\end{aligned}$$

so that the covariance can be determined as

$$\begin{aligned}C_u(\mathbf{x},\mathbf{y}) &= \mathbb{E}[u(\mathbf{x},\xi)u(\mathbf{y},\xi)] - \mathbb{E}[u(\mathbf{x},\xi)]\mathbb{E}[u(\mathbf{y},\xi)] \\ &= \sum_{n=1}^{\infty} u_n(\mathbf{x})u_n(\mathbf{y})n!.\end{aligned}$$

The Generalized Polynomial Chaos

The idea of polynomial chaos can be generalized for any distribution of random variables. Let $(\Omega, \mathcal{A}, \mathbb{P})$ be a probability space and $\xi \in \Omega$ a random variable with the density function ρ. Instead of Hermite polynomials we expand a second order random function $u \in L^2(\Omega, \rho)$ in terms of the corresponding orthogonal basis polynomials ψ_k. Table 3.2 gives a short overview over some densities with corresponding orthogonal polynomials. The generalized polynomial chaos (gPC) expansion is given by

$$u(\xi) = \sum_{k=0}^{\infty} u_k \psi_k(\xi) \tag{3.22}$$

with coefficients

$$u_k = \frac{\langle u, \psi_k\rangle}{\langle \psi_k, \psi_k\rangle}.$$

As in the classical case, the generalized polynomial chaos expansion converges in $L^2(\Omega, \rho)$ and minimizes the root mean square error in the space of all polynomials up to degree N. Moreover, the convergence is again spectral, see [Xiu10]. Also, the generalized polynomial chaos expansion decouples space and randomness and the statistics of u can be calculated analogously, i.e.

$$\begin{aligned}\mathbb{E}[u(\mathbf{x},\xi)] &= u_0(\mathbf{x}), \\ \mathrm{Var}[u(\mathbf{x},\xi)] &= \sum_{k=1}^{\infty} u_k^2(\mathbf{x})\langle \psi_k, \psi_k\rangle, \\ C_u(\mathbf{x},\mathbf{y}) &= \sum_{k=1}^{\infty} u_k(\mathbf{x})u_k(\mathbf{y})\langle \psi_k, \psi_k\rangle.\end{aligned}$$

Distribution	Polynomials	Domain
Gaussian	Hermite	$[-\infty, \infty]$
Uniform	Legendre	$[-1, 1]$
Beta	Jacobi	$[-1, 1]$
Gamma	Laguerre	$[0, \infty]$

Table 3.2: Different types of basis functions with corresponding densities.

The Multivariate Polynomial Chaos

The generalized polynomial chaos expansion (3.22) can be easily extended to multiple dimensions. Consider an $\mathbb{R}^d$-valued random vector $\boldsymbol{\xi} = (\xi_1, \dots, \xi_d)$ with independent components, support $\Omega = \Omega_1 \times \cdots \times \Omega_d$, and joint density $\boldsymbol{\rho} = \rho_1 \otimes \cdots \otimes \rho_d$. Let $\psi_k^{(i)}$ denote the k-th basis polynomial corresponding to the distribution of ξ_i, then the basis polynomial of multi-index $\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_d) \in \mathbb{N}_0^d$ is defined by the tensor product

$$\psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}) = \prod_{i=1}^{d} \psi_{\alpha_i}^{(i)}(\xi_i).$$

Hence, we obtain the d-dimensional polynomial chaos expansion

$$u(\mathbf{x}, \boldsymbol{\xi}) = \sum_{\boldsymbol{\alpha}} u_{\boldsymbol{\alpha}}(\mathbf{x}) \psi_{\boldsymbol{\alpha}}(\boldsymbol{\xi}).$$

Note that the components ξ_i of the random vector do not need to follow the same distribution.

Due to the tensor product ansatz and the decoupling of spatial and random dimensions the contribution from each uncertain input parameter can be easily identified making the multivariate polynomial chaos expansion an important tool in sensitivity analysis, see [Sud08, CMM09].

Example 3.32. Consider a random vector $\boldsymbol{\xi} = (\xi_1, \xi_2)$ with a standard Gaussian random variable $\xi_1 \sim \mathcal{N}(0, 1)$ and a uniformly distributed random variable $\xi_2 \sim \mathcal{U}([-1, 1])$. The univariate orthogonal polynomials for the standard Gaussian random variable are Hermite polynomials, and the univariate orthogonal polynomials for the uniformly distributed random variable are Legendre polynomials

$$L_n(\xi) = \frac{1}{2^n n!} \frac{\mathrm{d}}{\mathrm{d}\xi^n} (\xi^2 - 1)^n.$$

Hence, the bivariate orthogonal polynomials for $\boldsymbol{\xi}$ up to total degree 2 are given by

$$\begin{aligned}
\psi_{(0,0)} &= H_0 L_0 \\
&= 1, \\
\psi_{(1,0)} &= H_1 L_0 \qquad \psi_{(0,1)} = H_0 L_1 \\
&= \xi_1, \qquad\qquad = \xi_2, \\
\psi_{(2,0)} &= H_2 L_0 \qquad \psi_{(1,1)} = H_1 L_1 \qquad \psi_{(0,2)} = H_0 L_2 \\
&= \xi_1^2 - 1, \qquad\quad = \xi_1 \xi_2, \qquad\qquad = \tfrac{1}{2}(3\xi_2^2 - 1).
\end{aligned}$$

In general, truncation of a d-variate generalized polynomial chaos expansion after all polynomials with a total degree of p leads to a total number of N coefficients with

$$N + 1 = \frac{(d+p)!}{d!p!}.$$

Hence, the total number of coefficients grows combinatorially with the dimensionality d of the random input and the degree p of polynomial approximation. This rapid growth makes the generalized polynomial chaos expansion useless for practical applications with a large d or p. In Darcy flow problems where the dimension is small, i.e. $d = 2$ or $d = 3$, polynomial chaos expansion is often used for discretizing the solution space, see [BD17, TI14, AV13].

3.3.3 The Galerkin Projection

As in [MK10] we formulate the stochastic Galerkin method in an abstract way. For this let $\mathcal{M}$ be a mathematical model of a physical system specified by some data D. The data can describe the geometry, model parameters, boundary, and initial conditions. Then we search for a solution u of

$$\mathcal{M}(u, D) = 0. \tag{3.23}$$

Various problems can be represented with this abstract formulation, e.g. systems of ordinary differential equations, partial differential equations, algebraic equations, differential algebraic equations, or integral equations. In this formulation, M involves the full set of equations that are fulfilled by solution u, including potential boundary or initial conditions, and other constraints. To quantify the impact of uncertainty in the data on the solution u, consider a probability space $(\Omega, \mathcal{A}, \mathbb{P})$ with a sample space Ω, a σ-algebra $\mathcal{A}$, and a probability measure $\mathbb{P}$. Then we write $D(\omega)$ to indicate the dependence on the outcome $\omega \in \Omega$. Since the data is now random, the solution u is also random. Hence we aim for a solution of

$$\mathcal{M}(u(\omega), D(\omega)) = 0, \tag{3.24}$$

the stochastic version of (3.23). We assume that the mathematical model $\mathcal{M}(\cdot, D(\omega))$ has almost surely a unique solution in a suitable Hilbert space.

Now consider a truncated polynomial chaos expansion of the solution u

$$u(\mathbf{x}, \xi) = \sum_{k=0}^{N} u_k(\mathbf{x}) \psi_k(\xi)$$

where $\xi = \xi(\omega)$ and ψ_k are chosen appropriate to the random input. Substituting the polynomial chaos expansion into the original problem description yields

$$\mathcal{M}\left(\sum_{k=0}^{N} u_k(\mathbf{x})\psi_k(\xi), D(\xi)\right) = 0.$$

The density function of ξ defines a weight function for an inner product. Using this inner product we take a Galerkin projection of the original equation onto each basis polynomial ψ_k and obtain the weak formulation

$$\left\langle \mathcal{M}\left(\sum_{i=0}^{N} u_i(\mathbf{x})\psi_i(\xi), D(\xi)\right), \psi_k \right\rangle = 0, \quad k = 0, \ldots, N.$$

Hence, the stochastic approximation space is the same for ansatz and test functions.

Example 3.33. Consider the first-order ordinary differential equation

$$\dot{u}(t) = -\lambda u(t) \tag{3.25}$$

with initial conditions $u(0) = b > 0$ and a random parameter $\lambda > 0$. Let ψ_k be the orthogonal polynomials corresponding to the distribution of λ, then λ has the truncated polynomial chaos expansion

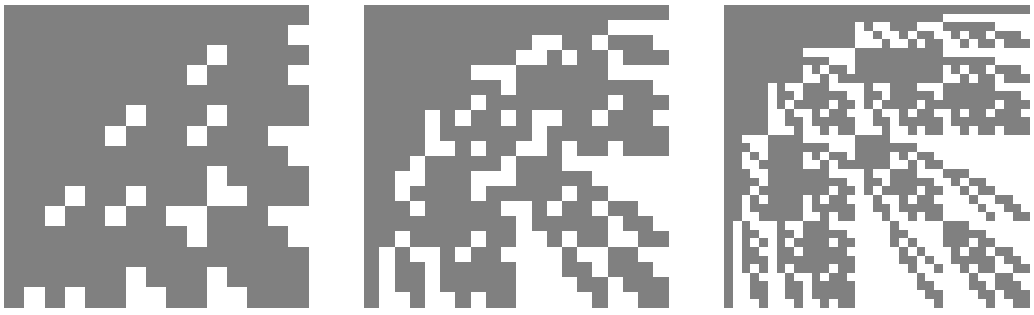
$$\lambda(\xi) = \sum_{k=0}^{N} \lambda_k \psi_k(\xi),$$

and we want to find the polynomial chaos expansion of the solution

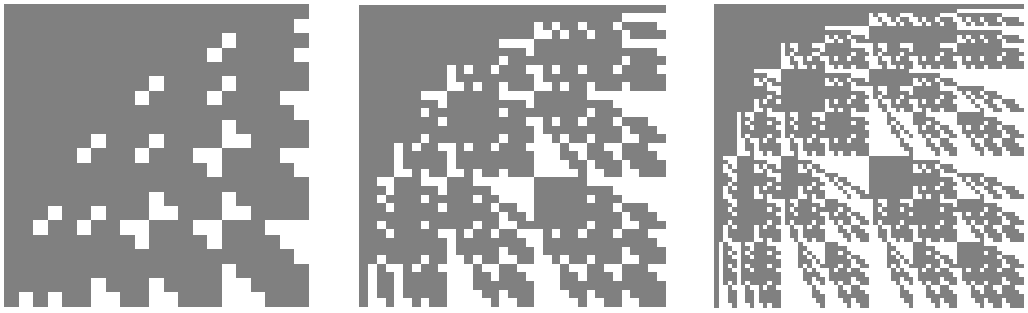
$$u(t, \xi) = \sum_{k=0}^{N} u_k(t)\psi_k(\xi).$$

The Galerkin projection of (3.25) and substitution of the polynomial chaos expansions yields

$$\begin{aligned}
\left\langle \sum_{j=0}^{N} \dot{u}_j(t)\psi_j, \psi_k \right\rangle &= -\left\langle \sum_{i=0}^{N} \lambda_i \psi_i \sum_{j=0}^{N} u_j(t)\psi_j, \psi_k \right\rangle \\
\dot{u}_k(t)\langle \psi_k, \psi_k \rangle &= -\sum_{i,j=1}^{N} \lambda_i u_j(t) \langle \psi_i \psi_j \psi_k \rangle \\
\dot{u}_k(t) &= -\sum_{i,j=1}^{N} \lambda_i u_j(t) \frac{\langle \psi_i \psi_j \psi_k \rangle}{\langle \psi_k \psi_k \rangle} \\
\dot{u}_k(t) &=: -\sum_{i,j=1}^{N} \lambda_i u_j(t) M_{ijk}.
\end{aligned}$$



(a) $d = 2$, $p = 3$ (b) $d = 3$, $p = 3$ (c) $d = 4$, $p = 3$



(d) $d = 2$, $p = 4$ (e) $d = 3$, $p = 4$ (f) $d = 4$, $p = 4$

Figure 3.8: The matrix $\mathbf{A}(\lambda)$ for different dimensions and truncation levels. The zero entries are white and the non-zero entries are gray.

Hence, we obtain a coupled system of $N + 1$ ordinary differential equations. In matrix-vector form the vector $\mathbf{u}(t) = (u_0(t), \ldots, u_N(t))$ of coefficients is the solution of

$$\dot{\mathbf{u}}(t) = -\boldsymbol{A}(\lambda)\mathbf{u}(t), \quad \mathbf{u}(0) = \mathbf{b},$$

where the matrix $\mathbf{A}(\lambda) \in \mathbb{R}^{(N+1)\times(N+1)}$ has the entries $a_{k,i} = \sum_{j=0}^{N} \lambda_i M_{ijk}$. Note, that the matrix $\mathbf{A}(\lambda)$ is sparse due to the orthogonality of the basis polynomials ψ_k and can be evaluated once and stored for any given number of expansion terms and dimensions. See Figure 3.8 for the structure of $\mathbf{A}(\lambda)$ for dimensions $d = 2, 3$ and a truncation after a total polynomial degree of $p = 3, 4, 5$. Zero entries are white and non-zero entries are gray. As you can see the sparsity increases with increasing dimension, for $d = 2$ approximately 13%, for $d = 3$ already about 26%, and for $d = 4$ about 37% of the entries are zero.

When solving a diffusion equation with an uncertain diffusion coefficient, often all previously discussed methods of spectral expansions are incorporated, see [TMEP11, BTNT12, BNT07]. Typically, a diffusion coefficient is approximated with a truncated Karhunen-Loève expansion which decorrelates the process. Then the problem is solved by taking a Galerkin projection onto the orthogonal polynomials with respect to the distribution of the uncorrelated random variables resulting from the Karhunen-Loève expansion. Hence, the solution of a diffusion equation is a polynomial chaos expansion in terms of these orthogonal polynomials.

3.3.4 Non-Intrusive Spectral Projection Methods

In the previous section we have seen how to reformulate the original problem in order to determine the coefficients of a polynomial chaos expansion. In contrast, we now introduce another possibility to determine the coefficients which involves only multiple realizations of the original problem. These realizations are used in a quadrature rule to obtain an approximate orthogonal projection, see [EB09, KW16]. Because of that, these methods are also called *non-intrusive spectral projection* (NISP).

Consider again a second-order random process $u \in L^2(\Omega, \rho)$ with a spectral expansion $u(\xi) = \sum_{k=1}^{\infty} u_k \psi_k(\xi)$ and coefficients as usual

$$u_k = \frac{\langle u, \psi_k \rangle}{\langle \psi_k, \psi_k \rangle} = \frac{1}{\langle \psi_k, \psi_k \rangle} \int_\Omega u(\xi)\psi_k(\xi)\rho(\xi)\,\mathrm{d}\xi. \tag{3.26}$$

At least one needs to approximate the integral with respect to ρ of the product of u with each basis function ψ_k, since the normalization constants $\langle \psi_k, \psi_k \rangle$ are often known in advance. Sometimes, also the normalization constants must be approximated. In any case, we use realizations of u to approximate the coefficients $u_k \approx \tilde{u}_k$. The resulting process

$$\tilde{u}(\xi) := \sum_{k=1}^{\infty} \tilde{u}_k \psi_k(\xi) \approx u(\xi) \tag{3.27}$$

is called *surrogate* of the original process u.

For example, one application of non-intrusive spectral projection methods are isothermal, isobaric molecular dynamics simulations of water at ambient conditions with parametric uncertainty. In this case Gaussian quadrature rules are used for integration, see [RND$^+$12a, RND$^+$12b].

Quadrature Methods

In principal, we can use any quadrature rule from Section 3.2 to approximate the coefficients. If the dimensionality of ξ is low and $u(\xi)$ is a smooth function of ξ, it is obvious to use a deterministic quadrature rule. An optimal polynomial accuracy will be reached with Gaussian quadrature using the roots of the orthogonal basis polynomials as quadrature points. In this case, the normalization constants $\langle \psi_k, \psi_k \rangle$ can be exactly computed with at least $(k+1)/2$ nodes. A drawback of Gaussian quadrature is that the nodes are not nested and past realizations cannot be reused for a more accurate quadrature rule. Therefore, one might prefer nested quadrature formulas such as the Clenshaw-Curtis quadrature. Again, sparse grid quadrature rules can overcome the curse of dimensionality up to a certain extent. If the dimensionality of ξ is high or u is non-smooth, it makes sense to use a Monte Carlo quadrature approximation of $\langle u, \psi_k \rangle$. But again, the convergence rate of $\mathcal{O}(-1/2)$ is very slow. For smoother random fields u, the rate can be improved by using quasi-Monte Carlo methods.

The Connection with Linear Least Squares

The determination of approximate spectral coefficients $\tilde{u}_k$ is related to a least-squares minimization, cf. [Sul15]. Let $\mathbf{V}$ be the Vandermonde matrix of basis functions ψ_k and quadrature points ξ_i, i.e.

$$\mathbf{V} := \begin{pmatrix} \psi_0(\xi_0) & \dots & \psi_N(\xi_0) \\ \vdots & \ddots & \vdots \\ \psi_0(\xi_N) & \dots & \psi_N(\xi_N) \end{pmatrix}.$$

Moreover, let $Q(u) := \sum_{i=1}^m w_i u(\xi_i)$ be an m-point quadrature rule with weights w_i and nodes ξ_i and let $\mathbf{W} := \operatorname{diag}(w_1, \dots, w_m)$. Now consider the surrogate (3.27) and denote by $u(\xi_i)$ the observations of the random field u at points ξ_i. The residuals r_i are defined as the distances between observations and predictions of the surrogate, i.e.

$$r_i := u(\xi_i) - \tilde{u}(\xi_i).$$

Theorem 3.34. *Let $\mathbf{u} = (u(\xi_1), \dots, u(\xi_m))$ be the vector of observations of the true model and $\tilde{\mathbf{u}} = (\tilde{u}_0, \dots, \tilde{u}_N)$ the vector of coefficients of the surrogate $\tilde{u} = \sum_{k=1}^N \tilde{u}_k \psi_k(\xi)$. Then the following statements are equivalent:*

1. *$\tilde{u}$ minimizes the weighted sum of residuals*

$$\sum_{i=1}^m w_i r_i^2 = \sum_{i=1}^m w_i \big(u(\xi_i) - \tilde{u}(\xi_i)\big)^2, \tag{3.28}$$

2. *$\tilde{\mathbf{u}}$ satisfies*

$$(\mathbf{V}^T \mathbf{W} \mathbf{V}) \tilde{\mathbf{u}} = \mathbf{V}^T \mathbf{W} \mathbf{u},$$

3. *$\tilde{u} = u$ in the weak sense, tested against the basis functions ψ_k using the quadrature rule Q, i.e. for $k = 0, \dots, N$*

$$Q(\psi_k \tilde{u}) = Q(\psi_k u). \tag{3.29}$$

Proof. By the definition of the polynomial chaos expansion, we have

$$\mathbf{V}\tilde{\mathbf{u}} = \begin{bmatrix} \tilde{u}(\xi_1) \\ \vdots \\ \tilde{u}(\xi_m) \end{bmatrix}.$$

Hence, the weighted sum of residuals $\sum_{i=1}^m w_i (\tilde{u}(\xi_i) - u(\xi_i))^2$ can be written as

$$\begin{aligned} \|\mathbf{V}\tilde{\mathbf{u}} - \mathbf{u}\|_{\mathbf{W}}^2 &= (\mathbf{V}\tilde{\mathbf{u}} - \mathbf{u})^T \mathbf{W} (\mathbf{V}\tilde{\mathbf{u}} - \mathbf{u}) \\ &= \tilde{\mathbf{u}}^T \mathbf{V}^T \mathbf{W} \mathbf{V} \tilde{\mathbf{u}} - 2\mathbf{u}^T \mathbf{W} \mathbf{V} \tilde{\mathbf{u}} + \mathbf{u}^T \mathbf{W} \mathbf{u}. \end{aligned}$$

It is always possible to find a solution that minimizes the least squares problem. If the matrix $\mathbf{V}$ has full rank, the solution is unique. Differentiation with respect to the $\tilde{u}_k$'s yields

$$\nabla \|\mathbf{V}\tilde{\mathbf{u}} - \mathbf{u}\|_{\mathbf{W}}^2 = 2(\mathbf{V}\tilde{\mathbf{u}} - \mathbf{u})^T \mathbf{W}\mathbf{V},$$

and hence $\tilde{\mathbf{u}}$ is a minimizer if, and only if, it satisfies the normal equations

$$(\mathbf{V}^T\mathbf{W}\mathbf{V})\tilde{\mathbf{u}} = \mathbf{V}^T\mathbf{W}\mathbf{u}$$

Therefore 1. $\Leftrightarrow$ 2. Next, calculation of the left- and right-hand sides of (3.29) yields

$$\sum_{i=1}^{m} w_i \begin{bmatrix} \psi_0(\xi_i)\tilde{u}(\xi_i) \\ \vdots \\ \psi_k(\xi_i)\tilde{u}(\xi_i) \end{bmatrix} = \sum_{i=1}^{m} w_i \begin{bmatrix} \psi_0(\xi_i)u(\xi_i) \\ \vdots \\ \psi_k(\xi_i)u(\xi_i) \end{bmatrix},$$

which shows by comparison of the terms that 2. $\Leftrightarrow$ 3. □

Note that if the quadrature rule Q is a Gaussian or Monte Carlo quadrature rule appropriate to the measure μ, the matrix $\mathbf{V}^T\mathbf{W}\mathbf{V}$ is an approximation to the Gram matrix of the basis functions $\psi_0, \ldots, \psi_k$ in the $L^2(\mu)$ inner product. In particular, we have

$$\tilde{u}_k \approx \frac{Q(\psi_k u)}{\langle \psi_k, \psi_k \rangle},$$

i.e. $\tilde{u}_k$ approximately satisfies the orthogonal projection condition (3.26) satisfied by the polynomial chaos coefficient u_k.

In practice, if the observation points ξ_i are not associated with some quadrature rule for μ, one constructs an approximate polynomial chaos expansion by choosing the coefficients $\tilde{u}_k$ to minimize the weighted sum of residuals (3.28). In contrast, one can select the points ξ_i so that they optimize some quantity of the matrix $\mathbf{V}$, cf. [GI17, Sul15]. Common choices are for example:

- A-optimality: minimize the trace $(\mathbf{V}^T\mathbf{V})^{-1}$
- D-optimality: maximize the determinant $\mathbf{V}^T\mathbf{V}$
- E-optimality: maximize the lower singular value of $\mathbf{V}^T\mathbf{V}$
- G-optimality: minimize the largest diagonal term in the orthogonal projection $\mathbf{V}(\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T$

3.4 Stochastic Collocation

Stochastic collocation is also a non-intrusive solution method that incorporates several solutions of the original problem. In contrast to non-intrusive spectral methods, the approximation is constructed not on a pre-defined stochastic subspace but

instead is based on interpolation. Hence, the approximation as well as the approximation space are implicitly defined by the collocation nodes x_i. As the number of points increases, the space over which the solution is sought becomes larger. In stochastic collocation methods, one usually seeks for a polynomial approximation $\tilde{f}(x)$ of $f(x)$ which is exact at the selected set of n collocation points, i.e.

$$\tilde{f}(x_i) = f(x_i), \quad i = 1, \dots, n. \tag{3.30}$$

Because the approximation is exact on a finite set of points, the method is called *collocative*. Under the assumption that the collocation points are distinct, we can construct an approximation space of dimension n. Let ϕ_k denote corresponding basis functions, then the approximation of f is given by the expansion

$$\tilde{f}(x) = \sum_{k=1}^{n} \tilde{f}_k \psi_k(x), \tag{3.31}$$

where the coefficients $\tilde{f}_k$ are determined from the constraints (3.30). Obviously, the constraints are fulfilled if the basis functions have the following properties

$$\psi_k(\xi_i) = \delta_{i,k}, \quad 1 \leq i, k \leq n. \tag{3.32}$$

Hence, we obtain for the coefficients

$$\tilde{f}_i = f(x_i), \quad 1 \leq i \leq n.$$

Note that this basis is orthogonal only for a particular set of collocation points and hence (3.31) is generally not an orthogonal expansion with respect to the inner product of $L^2(\Omega, \rho)$. This is in contrast to the polynomial chaos expansion which is an orthogonal expansion with respect to the inner product.
In the following we discuss how to define and construct basis functions or interpolation functions of the form (3.31)–(3.32). There are different possibilities, the most common way is polynomial interpolation. Additionally, we can distinguish between two types of support of the basis function ψ_k. Either the support is the entire domain Ω or the support of ψ_k is only a sub-domain $\Omega_k \subset \Omega$. For the latter we will obtain a piecewise polynomial approximation.

3.4.1 The Lagrange Interpolation

Consider a function $f(x)$ and a set of $n+1$ distinct collocation points x_i. We seek a polynomial $p \in \mathcal{P}_n$ of degree at most n which interpolates f, i.e.

$$p(x_i) = f(x_i).$$

Such a polynomial p always exists and is unique. It is common to express p in the Lagrange basis (3.6) associated to the collocation points x_i, i.e.

$$\ell_i(x) = \prod_{\substack{0 \leq k \leq n \\ k \neq i}} \frac{x - x_k}{x_i - x_k}.$$

Obviously, the Lagrange basis polynomials fulfill (3.32) and therefore the interpolation polynomial p can be written as

$$p(x) := L_n(x) = \sum_{i=0}^{n} u(x_i)\,\ell_i(x).$$

In the following we denote by $\mathcal{I}$ the interpolation formula $\mathcal{I}(f) = p$. The next theorem helps us to estimate the interpolation error:

Theorem 3.35. *If f is $n+1$ times continuously differentiable on a closed interval, i.e. $f \in \mathcal{C}^{n+1}([a,b])$, and p is the polynomial of degree at most n that interpolates u at $n+1$ distinct points $x_i \in [a,b]$, then for each $x \in [a,b]$ there exists a $\xi \in [a,b]$ such that*

$$f(x) - p(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{i=0}^{n} (x - x_i).$$

Hence, we can estimate the interpolation error by

$$|f(x) - p(x)| \leq \frac{\|f^{(n+1)}\|_\infty}{(n+1)!} \prod_{i=0}^{n} |x - x_i|. \tag{3.33}$$

This error bound motivates the choice of Chebyshev nodes as interpolation points x_i because they minimize the product $\prod_{i=0}^{n} |x - x_i|$. Moreover, Chebyshev points have the advantage of being nested. If we use instead equally spaced collocation points $x_i = a + ih$, $h = (b-a)/n$, the error is bounded as

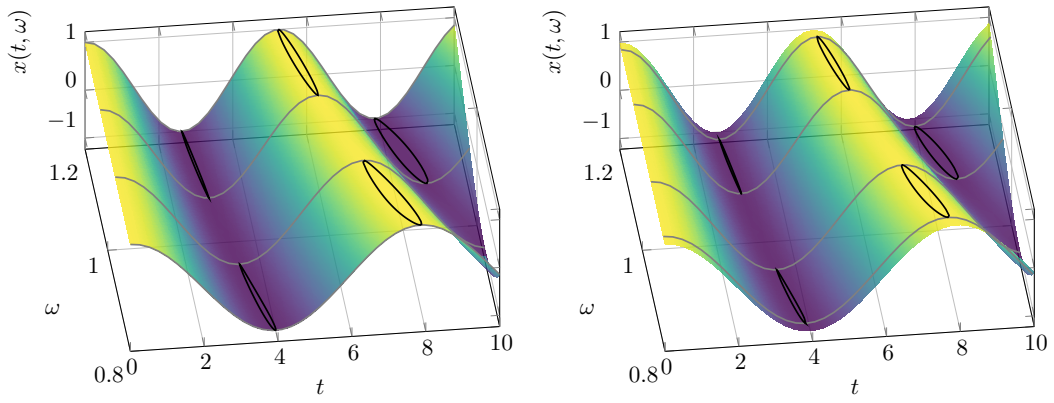
$$|f(x) - p(x)| \leq \frac{\|f^{(n+1)}\|_\infty}{(n+1)!} h^{n+1}.$$

Therefore, if $\|f^{(n+1)}\|_\infty$ is bounded on $[a,b]$, the interpolation $p(x)$ will converge to $f(x)$ as $n \to \infty$. But the assumption that $\|f^{(n+1)}\|_\infty$ is bounded on $[a,b]$ is not always fulfilled, see Runge's phenomenon in Section 3.2.1.

Example 3.36. Consider the harmonic oscillator with random frequency $\omega \sim \mathcal{U}([0.8, 1.2])$. The initial value problem reads

$$\begin{aligned} \ddot{x}(t) &= -\omega^2 x(t) \\ x(0) &= 1 \\ \dot{x}(0) &= 0, \end{aligned}$$

and its solution is $x(t) = \cos(\omega t)$. See Figure 3.9 for the interpolation with four equidistant and four Chebyshev nodes. Both surface plots are optically indistinguishable. Note that in both cases the interpolation yields unphysical values of $|x(t,\omega)| > 1$, but in the case of Chebyshev nodes the regions with unphysical values are smaller.



(a) Equidistant nodes.

(b) Chebyshev nodes.

Figure 3.9: The interpolation solutions of harmonic oscillator (Example 3.36) with four collocation points, respectively. The solutions for collocation points are shown in gray, contour lines in black depict $x(t,\omega) = 1$ and $x(t,\omega) = -1$.

3.4.2 Piecewise Polynomial Interpolation

Instead of approximating f by a global Lagrange interpolation, we can also use a piecewise continuous polynomial approximation. The easiest way is to approximate f by continuous linear polynomials. Consider a partition $a = x_0 < x_1 < \ldots < x_n = b$ of the support of f, then we approximate f with

$$p_i(x) = \begin{cases} f(x_i) + \dfrac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}(x - x_i), & x \in [x_i, x_{i+1}] \\ 0, & \text{else} \end{cases}$$

for $i = 0, \ldots, n-1$. Piecewise linear interpolation has the advantage that no unphysical values can arise because maximum and minimum of the interpolation are attained in the nodes. Moreover, the interpolations on the different intervals are independent of each other which allows to compute them parallely. Of course it is also possible to approximate f on each interval $[x_i, x_{i+1}]$ by polynomials of higher degree $k > 1$. For this we interpolate f in the $(k+1)$ nearest neighbors of the center $(x_i + x_{i+1})/2$. In this case the global approximation is continuous but not necessarily smooth, i.e. the derivatives of the approximation may have jumps at the collocation nodes.

In contrast, *spline interpolation* is an interpolation that is smooth everywhere, in particular at the collocation nodes. The classical spline interpolation is the *cubic* spline interpolation where polynomials of degree three are used. Originally, spline was a term for elastic rulers that were bent to pass through a number of predefined points. Under this constraint, the spline p takes a shape that minimizes the bending and both, the first and the second derivatives, are continuous everywhere, in particular at the nodes. This can only be achieved if polynomials of degree three or higher are used. Hence, we approximate the function f by the spline p which is a polynomial $p_i \in \mathcal{P}_3$ on subinterval $[x_i, x_{i+1}]$. Because each polynomial p_i has four

degrees of freedom we need $4n$ equations to solve the problem. Since p_i is continuous at nodes x_i, x_{i+1} we have the $2n$ equations

$$p_i(x_i) = f(x_i)$$
$$p_i(x_{i+1}) = f(x_{i+1})$$

for $i = 0, \ldots, n-1$. Moreover, we force the interpolation to be smooth at the interior nodes, which yields the $2(n-1)$ equations

$$p'_{i-1}(x_i) = p'_i(x_i)$$
$$p''_{i-1}(x_i) = p''_i(x_i)$$

for $i = 1, \ldots, n-1$. The two remaining equations are obtained by some boundary conditions, e.g. the natural boundary conditions $p''_0(x_0) = 0$ and $p''_{n-1}(x_n) = 0$ or the periodic boundary conditions $p'_0(x_0) = p'_{n-1}(x_n)$ and $p''_0(x_0) = p''_{n-1}(x_n)$. The natural boundary conditions are motivated by the elastic rulers that can move freely to the left of the left-most node and to the right of the right-most node and therefore the ruler will take on the form of a straight line with no bending, i.e. $p''(x) = 0$. In contrast to polynomial interpolation where the inflection points are located next to or in the nodes, spline interpolation results in inflection points between the nodes where the linearity is maximal and the bending minimal. At the nodes the curvature is maximal because of the maximal force acting on the spline by fixation.

For natural cubic splines it holds the following convergence statement, cf. [Ker71]:

Theorem 3.37. *Let $p(x)$ be a natural cubic spline, $f \in \mathcal{C}^4[a,b]$ and*

$$p(x_i) = f(x_i), \quad i = 0, 1, \ldots, n.$$

Then there exist nodes x_p, x_q for sufficiently large n, where

$$a < x_p < x_q < b,$$

and a constant c such that for $x_p \leq x \leq x_q$

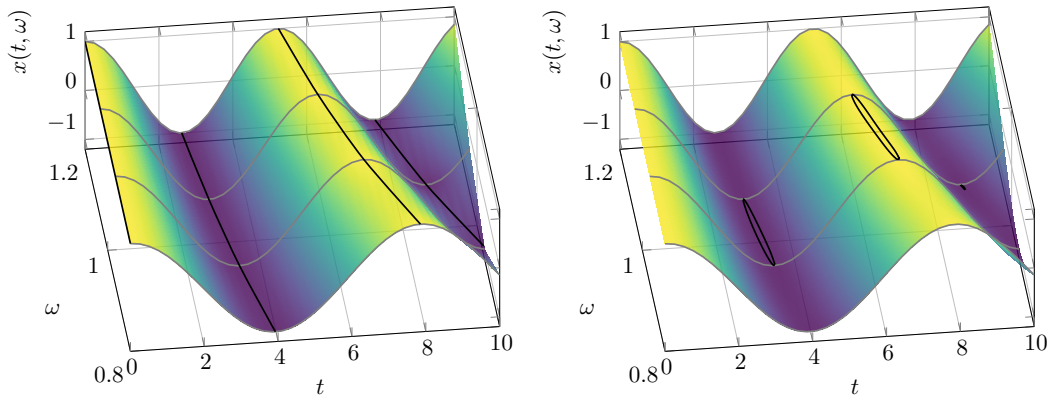
$$\begin{aligned} \max| \ f(x) - p(x) \ | &\leq \ ch^4 \\ \max| \ f'(x) - p'(x) \ | &\leq 4ch^3 \\ \max|f''(x) - p''(x)| &\leq 8ch^2. \end{aligned}$$

Further,

$$x_p - a = \mathcal{O}(h \log h), \quad b - x_q = \mathcal{O}(h \log h) \text{as } h \to 0, \tag{3.34}$$

where $h = \max|x_{i+1} - x_i|$.

This result shows that the convergence of the interpolating cubic spline is not uniformly $\mathcal{O}(h^4)$ as it is the case when Lagrange interpolation of degree four is used, see 3.33. But for a sufficiently large number of points it is $\mathcal{O}(h^4)$ except in the two end intervals, which tend to zero as the maximum interval h tends to zero.



(a) Piecewise linear interpolation.

(b) Cubic spline interpolation.

Figure 3.10: The interpolation solutions of harmonic oscillator (Example 3.36) with four collocation points, respectively. The solutions for collocation points are shown in gray, contour lines in black depict $x(t,\omega) = 1$ and $x(t,\omega) = -1$.

Example 3.38. Consider again example 3.36. Choosing a piecewise linear interpolation between the collocation nodes has the advantage that the interpolation yields no unphysical values $|x(t,\omega)| > 1$ and for each $\omega \in [0.8, 1.2]$ the extrema $x|(t,\omega)| = 1$ are attained, see Figure 3.10(a). Cubic spline interpolation results in smaller regions of unphysical values than Lagrange-interpolation although both use the same nodes and the same polynomial degree, see Figure 3.10(b).

3.4.3 Multivariate Interpolation

Analogously to multi-variate quadrature formulas, a d-dimensional interpolation formula can be constructed as a tensor product of one-dimensional interpolation formulas $\mathcal{I}^1$:

$$\begin{aligned} f(\mathbf{x}) &\approx (\mathcal{I}^1 \otimes \ldots \otimes \mathcal{I}^1)(f) \\ &= \sum_{i_1=1}^{n} \ldots \sum_{i_d=1}^{n} f(x_{i_1}, \ldots, x_{i_d}) L_{i_1}(x^{(1)}) \cdots L_{i_d}(x^{(d)}) \\ &=: \mathcal{I}^d(f) \end{aligned}$$

Both, multi-variate quadrature and multi-variate interpolation, suffer from the curse of dimensionality. This again motivates the use of Smolyak's algorithm [Smo63] to reduce the complexity of the collocation methods. Consider a sequence of one-dimensional interpolation formulas $\mathcal{I}_l^1$ on n_l^1 nested nodes. Then we define as in (3.15) the difference interpolation formula by

$$\Delta_k f := (\mathcal{I}_k^1 - \mathcal{I}_{k-1}^1) f$$

with $\mathcal{I}_0^1 f = 0$ and $\mathcal{I}_1^1 f = f(x_0)$. Then the sparse grid interpolation formula for d-dimensional functions f is given as

$$\mathcal{I}_l^d(f) := \sum_{|\mathbf{k}|_1 \leq l+d-1} (\Delta_{k_1} \otimes \ldots \otimes \Delta_{k_d}) f$$

for level $l \in \mathbb{N}$ and multi-index $\mathbf{k} \in \mathbb{N}^d$. Again, the number of nodes is of order $\mathcal{O}(2^l l^{d-1})$ if the one-dimensional interpolation formula has $\mathcal{O}(2^l)$ nodes.

To quantify the uncertainty in the solution of a diffusion equation with a random diffusion coefficient all types of sparse grids are used, standard [ES14, TMEP11], space adaptive [FP16], and dimension adaptive [BTNT12] sparse grids. Since the dimensionality is not so high in this application, also interpolation with Hermite or Legendre polynomials can be used on the full grid [BNT07].

If the nodes are unstructured and do not have a tensor product structure, collocation is not easy. In this case the existence of interpolating polynomials such as analogues of the Lagrange basis polynomials is not guaranteed. One possibility is to calculate the Delaunay triangulation defined by the set of nodes and approximate f by a piecewise linear interpolation on each simplex. In doing so, we obtain a continuous piecewise linear approximation of f. Also polynomials of higher degree can be used. This method is called *simplex stochastic collocation* and will be discussed in detail in the next chapter.

3.5 Stochastic Galerkin vs. Stochastic Collocation

In the previous sections we have seen several methods for uncertainty quantification but the most interesting question is when to use which of them. The stochastic Galerkin method is able to deal with a steep non-linear dependence of the solution on the random input and if the solution is sufficiently smooth an exponential convergence can be obtained. Moreover, the Galerkin projection minimizes the stochastic residual which means that the Galerkin method has optimal accuracy. But the stochastic Galerkin method has also some major drawbacks. The original deterministic system must be modified which results in a larger system of coupled equations. The properties of the coupled system may not be clear even if the deterministic system was simple. Moreover, the original code cannot be used and must be modified. If we have a deterministic solver we want to use, stochastic collocation is the method of choice since it is non-intrusive. Because the system is decoupled all simulation runs are independent from each other and can be done in parallel. Stochastic collocation is not affected by the complexity of the original problem so long a solver for the deterministic problem exists. If the solution is sufficiently smooth in the random dimensions also stochastic collocation methods achieve a fast convergence. Another advantage is that the solution space is not pre-defined. The more points are used for the collocation the larger the solution space becomes. But stochastic collocation introduces aliasing errors because of the quadrature or interpolation scheme. The higher the dimension of the random space the more significant these errors can become. For a fixed accuracy measured in terms of the polynomial exactness of the

approximation, collocation methods require the solution of a much larger number of equations than that required by a polynomial chaos Galerkin method. So if the simulations are very time-consuming and the coupling is not to complicated one should use the Galerkin projection because the least number of equations is needed.

4 Simplex Stochastic Collocation

In the simulation of gas networks the solution functions are continuous but not globally differentiable due to human intervention through the use of control valves, compressors, or heaters. Kinks in a function arise at hyper-surfaces where the function is not continuously differentiable. Since there is an existing solver that we want to use, we need an appropriate non-intrusive method for uncertainty quantification. In this application the quantities of interest are statistics of the solution, such as the expectation value, the variance, the median, or the cumulative density function of the solution. Of course, all these statistics can be computed with Monte Carlo methods, but this would result in a poor convergence rate of $\mathcal{O}(-\frac{1}{2})$, and many time consuming simulations would be needed to obtain a sufficiently small error. Although there are many non-intrusive methods available for uncertainty quantification – see the previous discussed methods, such as quasi-Monte Carlo methods, Gauss quadrature, Newton Cotes quadrature, standard sparse grid quadrature and interpolation, or non-intrusive spectral methods – none of them are useful for gas networks because all of them require sufficiently smooth functions for a fast convergence. Suitable techniques for non-smooth functions are very rare. Only spatially adaptive sparse grids or simplex stochastic collocation (SSC) [WI12a, WI12b, WI13] can handle discontinuities by adaptively placing more points in the non-smooth regions.

The idea of simplex stochastic collocation is to approximate a function f by a piecewise polynomial interpolation on simplices. Since polynomial interpolation gets oscillatory near discontinuities, one ensures that the approximation is local extremum conserving, i.e. maximum and minimum of the approximation in any simplex must be attained at its vertices, otherwise the polynomial degree is decreased by one. This condition results in a fine discretization near discontinuities and a coarser discretization at smooth regions. We tested the original approach for functions with kinks but were not able to reach the desired convergence rates because by increasing the polynomial degree the approximation of a kink cannot be improved. Based on the original simplex stochastic collocation, we developed a new approach by taking advantage of some special knowledge in gas network simulation. Of course, we have no information regarding the location of the kink, but we know which elements of the gas network cause kinks. After simulating the gas flow for a specific combination

of uncertain parameters, we know whether the kink inducing elements are active or not. In the case of a control valve we only need to check if the outgoing pressure lies below the preset pressure p_{set} or equals it. Therefore we know on which side of the kink the current simulation is located. This additional information enables us to approximate the function on each side of the kink separately. In doing so we can improve the convergence rate significantly without wasting sampling points near the kink.

4.1 Function Approximation

Following the approach of simplex stochastic collocation after [WI12a, WI12b, WI13] let $\Omega = [0,1]^d$ and $f : \Omega \to \mathbb{R}$ be a continuous function. The Delaunay triangulation of n uniformly distributed sampling points $\mathbf{x}_i$, which always include the corners of Ω, divides the parameter space Ω into m disjoint simplices T_j. Each simplex T_j is defined by its $d+1$ vertices $\mathbf{x}_{i_{j,l}}$ with $i_{j,l} \in \{1, \ldots, n\}$ and $l \in \{0, \ldots, d\}$.

4.1.1 The Original SSC

Let $f \in \mathcal{C}^0(\Omega)$ be a continuous function that we approximate by m piecewise polynomial functions $g_j(\mathbf{x})$ defined on simplex T_j

$$f(\mathbf{x}) \approx \sum_{i=1}^{m} g_j(\mathbf{x}) \, \mathbb{1}_{T_j}.$$

The polynomials g_j are defined as

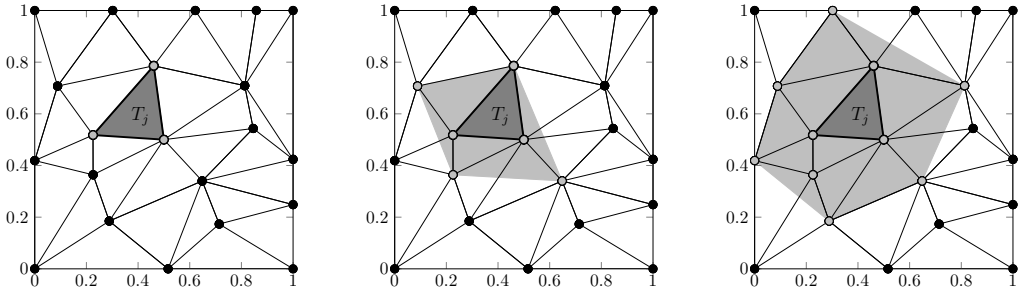
$$g_j(\mathbf{x}) = \sum_{k=1}^{N_j} c_{j,k} \psi_{j,k}(\mathbf{x}),$$

where $\psi_{j,k}$ are some appropriate basis polynomials, $c_{j,k}$ the corresponding coefficients, and $N_j = (d+p_j)!/(d!p_j!)$ the number of degrees of freedom with $p_j \leq p_{\max}$ the local polynomial degree. The polynomial approximation $g_j(\mathbf{x})$ in T_j is constructed by interpolating $f(\mathbf{x})$ in a stencil

$$S_j = \{\mathbf{x}_{i_{j,0}}, \ldots \mathbf{x}_{i_{j,N_j}}\}$$

consisting of N_j points out of the sampling points $\mathbf{x}_i$. These points are chosen to be the nearest neighbors to simplex T_j based on the Euclidean distance to its center of mass. Since in the case of long and flat simplices not necessarily all of its vertices belong to the set of nearest neighbors, we choose the $d+1$ simplex vertices as the first nearest neighbors. Thus, we ensure that our approximation is exact at all sampling points. See Figure 4.1 for different nearest neighbor stencils of simplex T_j corresponding to polynomial degrees $p_j = 1, 2, 3$. If the interpolation problem is not unique solvable we reduce the polynomial degree p_j successively by one until the solution is unique. To avoid oscillations in an approximation $g_j(\mathbf{x})$ near a discontinuity, the local polynomial degree p_j is also reduced by one if the approximation is not local extremum conserving (LEC), i.e. if it does not hold that

$$\min_{\mathbf{x} \in T_j} g_j(\mathbf{x}) = \min_{\mathbf{x}_i \in T_j} f(\mathbf{x}_i) \quad \wedge \quad \max_{\mathbf{x} \in T_j} g_j(\mathbf{x}) = \max_{\mathbf{x}_i \in T_j} f(\mathbf{x}_i).$$



(a) $p_j = 1$, 3-nn (b) $p_j = 2$, 6-nn (c) $p_j = 3$, 10-nn

Figure 4.1: Shown are the Delaunay triangulation of $n = 20$ sampling points and the nearest neighbor stencils S_j (light gray) for simplex T_j (dark gray) for polynomial degrees $p_j = 1, 2, 3$.

Note that the polynomial degree will be at least one since the linear interpolation problem on a simplex is always uniquely solvable and the resulting interpolation is always local extremum conserving. Since the approximation in one single simplex is independent from all other simplices, the resulting global approximation is not even continuous across the simplices' facets, except for linear polynomials.

The Theoretical Convergence Rate

For smooth functions $f \in \mathcal{C}^{p+1}$ we can estimate the approximation error. Let $\{\mathbf{x}_\alpha\}_{|\alpha|\leq p}$ denote the interpolation points with multi-index $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_d) \in \mathbb{N}_0^d$. The classic estimation [SX95] for the error in the d-dimensional point $\mathbf{x} = (x^{(1)}, \ldots, x^{(d)})$ between the function $f(\mathbf{x})$ and its Lagrange interpolation $L_p f(\mathbf{x})$ of degree p reads

$$|L_p f(\mathbf{x}) - f(\mathbf{x})| \leq \sum_{|\alpha|=p+1} \frac{1}{\alpha!} \left\| \frac{\partial^{p+1} f}{\partial \mathbf{x}^\alpha} \right\|_\infty \prod_{\gamma_1=1}^{\alpha_1} \left(x^{(1)} - x^{(1)}_{(\gamma_1-1,\alpha_2,\ldots,\alpha_d)} \right) \cdots \prod_{\gamma_d=1}^{\alpha_d} \left(x^{(d)} - x^{(d)}_{(\alpha_1,\alpha_2,\ldots,\gamma_d-1)} \right). \tag{4.1}$$

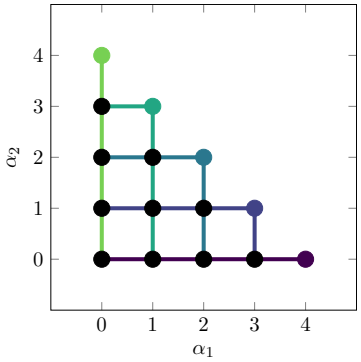
In the i-th product the i-th entry of α is replaced by $\gamma_i - 1$. Drawing n uniformly distributed random points in Ω, the expected distance between two of them is of order $\mathcal{O}(n^{-1/d})$. Because each summand consists of $p + 1$ factors, each summand is of order $\mathcal{O}(n^{-(p+1)/d})$. Thereby we can estimate the products in (4.1) and obtain

$$|L_p f(\mathbf{x}) - f(\mathbf{x})| \leq C \cdot n^{-(p+1)/d} \sum_{|\alpha|=p+1} \frac{1}{\alpha!} \left\| \frac{\partial^{p+1} f}{\partial \mathbf{x}^\alpha} \right\|_\infty. \tag{4.2}$$

Thus the Lagrange interpolation $L_p f$ converges pointwise with rate $\mathcal{O}(-(p+1)/d)$ against the function f if the partial derivatives are bounded. Because the convergence rate depends on the dimension we need to increase the polynomial degree with increasing dimension to obtain a constant convergence rate. The error estimation

(4.2) holds true for any simplex T_j and corresponding approximation $g_j(\mathbf{x})$. Note that for functions $f \in \mathcal{C}^0(\Omega)$ with kinks, i.e. functions that are continuous but not continuously differentiable, we cannot estimate the error with (4.2) or expect a convergence rate of $\mathcal{O}(-(p+1)/d)$, as $f \notin \mathcal{C}^{p+1}(\Omega)$. This motivates the modification of the original approach.

Example 4.1 (Terms in the Error Approximation)**.** Consider the case $d = 2$ and $p = 3$. Then we have the ten interpolation points numbered with the multi-indices $(0,0)$, $(1,0)$, $(0,1)$, $(2,0)$, $(1,1)$, $(0,2)$, $(3,0)$, $(2,1)$, $(1,2)$, and $(0,3)$. These indices are marked in black. Hence, in the error estimation, we sum over the five multi-indices $(4,0)$, $(3,1)$, $(2,2)$, $(1,3)$, and $(0,4)$, marked in different colors. All indices occurring in one summand are the indices corresponding to the black dots located on the line of the same color. Hence, the corresponding summands read



$$\frac{1}{4!}\left\|\frac{\partial^4 f}{\partial \mathbf{x}^{(4,0)}}\right\|_\infty \left(x^{(1)} - x^{(1)}_{(0,0)}\right)\left(x^{(1)} - x^{(1)}_{(1,0)}\right)\left(x^{(1)} - x^{(1)}_{(2,0)}\right)\left(x^{(1)} - x^{(1)}_{(3,0)}\right),$$

$$\frac{1}{4!}\left\|\frac{\partial^4 f}{\partial \mathbf{x}^{(3,1)}}\right\|_\infty \left(x^{(1)} - x^{(1)}_{(0,1)}\right)\left(x^{(1)} - x^{(1)}_{(1,1)}\right)\left(x^{(1)} - x^{(1)}_{(2,1)}\right)\left(x^{(2)} - x^{(2)}_{(3,0)}\right),$$

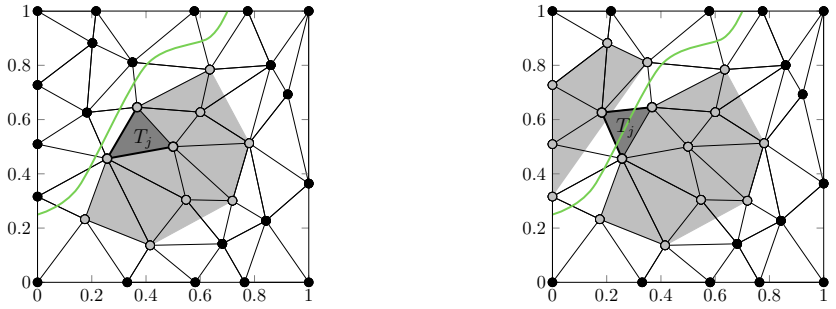
$$\frac{1}{4!}\left\|\frac{\partial^4 f}{\partial \mathbf{x}^{(2,2)}}\right\|_\infty \left(x^{(1)} - x^{(1)}_{(0,2)}\right)\left(x^{(1)} - x^{(1)}_{(1,2)}\right)\left(x^{(2)} - x^{(2)}_{(2,0)}\right)\left(x^{(2)} - x^{(2)}_{(2,1)}\right),$$

$$\frac{1}{4!}\left\|\frac{\partial^4 f}{\partial \mathbf{x}^{(1,3)}}\right\|_\infty \left(x^{(1)} - x^{(1)}_{(0,3)}\right)\left(x^{(2)} - x^{(2)}_{(1,0)}\right)\left(x^{(2)} - x^{(2)}_{(1,1)}\right)\left(x^{(2)} - x^{(2)}_{(1,2)}\right),$$

$$\frac{1}{4!}\left\|\frac{\partial^4 f}{\partial \mathbf{x}^{(0,4)}}\right\|_\infty \left(x^{(2)} - x^{(2)}_{(0,0)}\right)\left(x^{(2)} - x^{(2)}_{(0,1)}\right)\left(x^{(2)} - x^{(2)}_{(0,2)}\right)\left(x^{(2)} - x^{(2)}_{(0,3)}\right).$$

4.1.2 The Improved SSC

Let $f \in \mathcal{C}^0(\Omega)$ be a function with kinks. We say a function $f : [0,1]^d \to \mathbb{R}$ has a kink at the $(d-1)$-dimensional hyper-surface $K \subset \Omega$ if for all $\mathbf{x} \in K$ the function $f(\mathbf{x})$ is not continuously differentiable. In $d = 2$ dimensions, the kink locations are lines and can be arbitrarily shaped, they can be straight, curved or closed lines, and they can also intersect. In $d = 3$ dimensions, the kink locations are surfaces. They divide the parameter space Ω into disjoint subdomains Ω_k with $\bigcup_k \Omega_k = \Omega$. Suppose $f \in \mathcal{C}^{p+1}(\Omega_k)$ is smooth for all k, and that we have for each sampling point $\mathbf{x}_i$ the information to what Ω_k it belongs to. The last assumption is motivated by our application to gas networks where we also know if a regulator is active or not. Then there are two different cases for our modification:



(a) Simplex without a kink. (b) Simplex with a kink.

Figure 4.2: The improved nearest-neighbor stencils (light gray) for simplex T_j (dark gray). The domain Ω is divided by a kink (green) into two subdomains Ω_1 (left) and Ω_2 (right). (a) shown is a stencil for a simplex without a kink inside and completely lying in Ω_2. (b) shown are two stencils for a simplex with a kink inside and lying in Ω_1 as well as in Ω_2.

Case 1. For simplices T_j completely contained in some sub-domain Ω_k, i.e. $\exists k$ such that simplex $T_j \subset \Omega_k$, the function $f(\mathbf{x})$ is smooth. In this case we only search for the nearest neighbor stencil in the reduced set $\{\mathbf{x}_i | \mathbf{x}_i \in \Omega_k\}$ but not in the complete set of sampling points $\{\mathbf{x}_i\}$. As in the original approach, we ensure that the vertices $\mathbf{x}_{i_j}$ of simplex T_j are contained in the nearest neighbor stencil S_j. Since $S_j \subset \Omega_k$, we can approximate a smooth function by polynomial interpolation with known convergence rate $(p_j + 1)/d$. Figure 4.2(a) shows the improved stencil for a simplex T_j without any kinks inside.

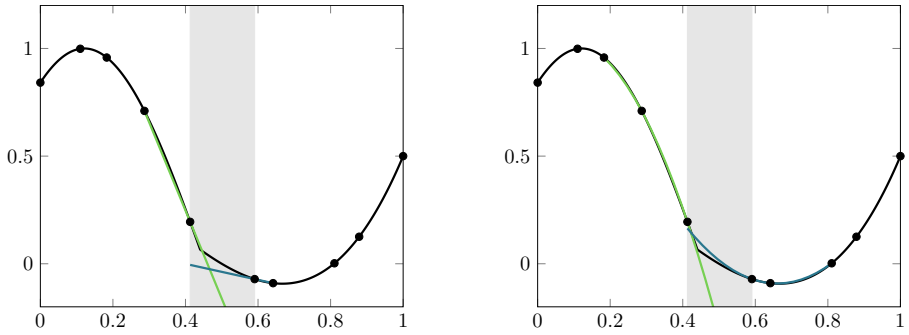
Case 2. Suppose simplex T_j is divided by a kink, i.e. some of its vertices $\mathbf{x}_{i_j}$ belong to Ω_{j_1} and some to Ω_{j_2}. In this case we search for two nearest neighbor stencils $S_{j,1} \subset \Omega_{j_1}$ and $S_{j,2} \subset \Omega_{j_2}$ and two approximations $g_{j,1}(\mathbf{x})$, $g_{j,2}(\mathbf{x})$, one at each side of the kink. As above we ensure that each stencil contains the corresponding vertices $\mathbf{x}_{i_j}$ of T_j. Without loss of generality we assume that the kink can be represented for all $\mathbf{x}_i \in S_{j,1} \cup S_{j,2}$ as the maximum of both interpolations, i.e.

$$f(\mathbf{x}_i) = \max\left(g_{j,1}(\mathbf{x}_i),\ g_{j,2}(\mathbf{x}_i)\right).$$

Then we extrapolate $g_{j,1}(\mathbf{x})$ and $g_{j,2}(\mathbf{x})$ to simplex T_j and approximate $f(\mathbf{x})$ for all $\mathbf{x} \in T_j$ by taking the maximum of both approximations

$$f(\mathbf{x}) \approx g_j(\mathbf{x}) := \max\left(g_{j,1}(\mathbf{x}),\ g_{j,2}(\mathbf{x})\right)$$

whereby we obtain an approximation to the kink. Figure 4.3 shows a linear and a quadratic approximation to a kink in simplex T_j. On both stencils S_{j_1} and S_{j_2} the function f is smooth and both approximations $g_{j,1}(\mathbf{x})$ and $g_{j,2}(\mathbf{x})$ converge with a rate of $(p_j + 1)/d$, respectively. Hence, the approximation $g_j(\mathbf{x})$ converges with the same rate. Note that this holds also true if $g_{j,1}(\mathbf{x})$ and $g_{j,2}(\mathbf{x})$ do not intersect in T_j. Even if there was not any kink in the function, this procedure of computing two approximations and taking the maximum would not affect the convergence. This is important because in our application an activated regulator, for example, may cause a kink in the flux in some pipes but not in all.



(a) Linear approximation. (b) Quadratic approximation.

Figure 4.3: A linear (a) and quadratic (b) approximation of a kink, each with two stencils. The function $f(x)$ is colored in black, the left hand approximation $g_{j,1}(x)$ in red and the right hand approximation $g_{j,2}(x)$ in blue.

4.2 Refinement Strategies

It is possible to construct an approximation for a given set of sampling points but we want to start with an initial set of sampling points consisting of the corners and the center of Ω, we then successively add new points at simplices with the largest error estimator to refine the discretization adaptively. In the end we aim for less points in regions were $f(\mathbf{x})$ is flat and more points in regions where $f(\mathbf{x})$ varies more. For an adaptive refinement we need on the one hand a strategy of how to add new points and on the other hand a reliable error estimator.

4.2.1 Adding a New Sampling Point

In [WI12b] simplex T_j is refined by sampling a new random point in a subsimplex T_{sub_j}. The vertices $\mathbf{x}_{\text{sub}_{j,l}}$ are defined as the centers of the faces of simplex T_j

$$\mathbf{x}_{\text{sub}_{j,l}} = \frac{1}{d} \sum_{\substack{l^*=0 \\ l^* \neq l}}^{d} \mathbf{x}_{i_{j,l^*}}$$

Due to [Dev86] an efficient way to sample uniform distributed random points in the unit simplex $S_d = \{(s_1, \ldots, s_d) : s_i \geq 0, \sum_{i=1}^{d} s_i \leq 1\}$ is the following: let $u_1, u_2, \ldots, u_{d+1}$ be independent and identically uniform in $[0, 1]$ distributed random numbers, then the random variables $e_1 = -\log(u_1), e_2 = -\log(u_2), \ \ldots, \ e_{d+1} = -\log(u_d)$ are independent and identically exponentially distributed with parameter $\lambda = 1$. Let $s = \sum_{i=1}^{d-1} s_i$, then the vector

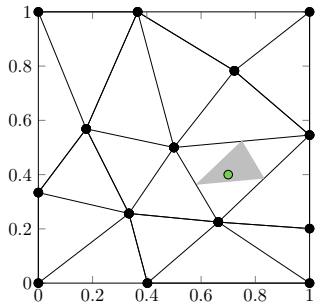
$$\mathbf{x} = (x_1, x_2, \ldots, x_d) = (e_1/s, e_2/s, \ldots, e_d/s)$$

is uniformly distributed in simplex S_d. This method has the advantage that sample points do not have to be rejected nor any numbers have to be sorted.
Figure 4.4(a) shows the subsimplex $T_{\text{sub}_j} \subset T_j$ of simplex T_j. This sampling strategy results in long and flat simplices at the boundary because the new sampling point

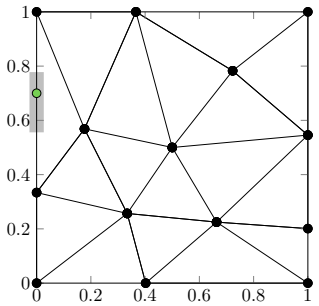
will almost surely not be added at the boundary. Therefore, we use this strategy only for simplices whose longest edge lies in the interior. Simplices whose longest edge lies at the boundary are refined by adding a new sampling point on the middle third of the longest edge as introduced in [WI12a]. Let $\mathbf{x}_{i_{j,0}}$ and $\mathbf{x}_{i_{j,1}}$ be the endpoints of the longest edge of simplex T_j, then we define the new sampling point $\mathbf{x}_{i_{\text{new}}}$ as

$$\mathbf{x}_{i_{\text{new}}} = \mathbf{x}_{i_{j,0}} + \tfrac{1+u}{3}\,(\mathbf{x}_{i_{j,1}} - \mathbf{x}_{i_{j,0}}),$$

where u is a uniformly distributed random variable in $[0, 1]$. Figure 4.4(b) shows the sampling area on the longest edge of a boundary simplex T_j.



(a) New sampling point (green) in subsimplex T_{sub_j} (light gray).



(b) New sampling point (green) on the longest edge at the boundary.

Figure 4.4: Shown are different refinement strategies for simplices with the longest edge lying in the interior of the domain Ω (a) and for simplices with the longest edge lying at the boundary $\partial\Omega$ (b).

4.2.2 Error Estimation

To refine the simplex with the largest error we need an error estimator since we cannot compute the exact error. First, we introduce two newly developed solution-based error estimators and then a third already existing error estimator that does not directly depend on the solution. The third one is very useful when the function $f(\mathbf{x})$ has more than one output.

Error Estimation Based on a Single Point

In [WI12a] a solution-based error estimator ε_j is proposed where the square of the hierarchical error $\epsilon_{i_{\text{new},j}} = |f(\mathbf{x}_{i_{\text{new},j}}) - g_j(\mathbf{x}_{i_{\text{new},j}})|$ between approximation and function at the new sampling point $\mathbf{x}_{i_{\text{new},j}}$ is weighted with the volume of the simplex $\varepsilon_j = \text{vol}(T_j) \cdot \epsilon^2_{i_{\text{new},j}}$. This error estimator has the disadvantage that we need to evaluate the function f at point $\mathbf{x}_{i_{\text{new},j}}$ although the point might not be added to the discretization. To avoid these useless function evaluations we modify the original error estimator and do not use the hierarchical error in the new sampling point $\mathbf{x}_{i_{\text{new},j}}$ but instead in the last added sampling point of simplex T_j before adding it. Let $i_{j^*} = \max_l i_{j,l}$ be the index of this last added sampling point and T_{ref,j^*} the

simplex which was refined by adding $\mathbf{x}_{i_{j^*}}$, then the hierarchical error is given by

$$\epsilon_{i_{j^*}} = |f(\mathbf{x}_{i_{j^*}}) - g_{\text{ref},j^*}(\mathbf{x}_{i_{j^*}})|$$

and we obtain the error estimator

$$\tilde{\varepsilon}_j = \text{vol}(T_j) \cdot \epsilon_{i_{j^*}}^2.$$

By summing up the error estimators for all simplices $\{T_j\}$ we can approximate the root mean square error in Ω by

$$\tilde{\varepsilon}_{\text{rms}} = \sqrt{\sum_{j=1}^{m} \tilde{\varepsilon}_j} = \sqrt{\sum_{j=1}^{m} \text{vol}(T_j) \cdot \epsilon_{i_{j^*}}^2}.$$

Error Estimation Based on Monte Carlo Integration

Because we do not want to rely on the error in one single point, we developed a new error estimator, an approximation of the L_1 error between approximation $g_j(\mathbf{x})$ and function $f(\mathbf{x})$ in a simplex T_j. For this we approximate $\varepsilon_j = \|f - g_j\|_{L_1(T_j)}$ by Monte Carlo integration, i.e.,

$$\varepsilon_j \approx \text{vol}(T_j) \sum_{i=1}^{n_{\text{MC}}} \frac{|f(\mathbf{x}_{\text{MC},i}) - g_j(\mathbf{x}_{\text{MC},i})|}{n_{\text{MC}}} \tag{4.3}$$

at n_{MC} randomly drawn Monte Carlo points $\mathbf{x}_{\text{MC},i}$. Usually it is not feasible to evaluate f at all n_{MC} Monte Carlo points because each function evaluation can be an expensive simulation. Thus we approximate the right hand side of (4.3) with the polynomial interpolation $\overline{g}_j$ in stencil S_j of degree $p_j - 1$.

$$\hat{\varepsilon}_j = \text{vol}(T_j) \sum_{i=1}^{n_{\text{MC}}} \frac{|g_j(\mathbf{x}_{\text{MC},i}) - \overline{g}_j(\mathbf{x}_{\text{MC},i})|^{(p_j+1)/p_j}}{n_{\text{MC}}}.$$

The exponent $(p_j + 1)/p_j$ is necessary since approximation $\overline{g}_j$ only leads to a convergence rate of p_j/d, whereas the approximation g_j converges with rate $(p_j + 1)/d$. Thereby we ensure that the error estimator decreases with the same rate as the true error. If $p_j = 1$, we define the constant function $\overline{g}_j$ as $\overline{g}_j(\mathbf{x}) = \min_{i_j} f(\mathbf{x}_{i_j})$. To obtain an overall error estimation, we sum up the error estimators for all simplices $\{T_j\}$

$$\hat{\varepsilon}_{l_1} = \sum_{j=1}^{m} \hat{\varepsilon}_j.$$

Error Estimation Based on the Theoretical Order of Convergence

If we have a function $f(\mathbf{x})$ with a multidimensional output, a solution-based error estimator could not be reasonably used because we do not know how the error scales over different outputs. Therefore we use the in [WI12a] described idea of a

solution-independent error estimator. For this consider the definition of the order of convergence

$$\mathcal{O} = \frac{\log(\varepsilon_0/\varepsilon_j)}{\log(\mathrm{vol}(\Omega)/\,\mathrm{vol}(\Omega_j))}$$

for some reference error ε_0. Then the error ε_j in simplex T_j is proportional to

$$\varepsilon_j \sim \mathrm{vol}(T_j)^{\mathcal{O}} = \mathrm{vol}(T_j)^{(p_j+1)/d}.$$

Weighting this again with the volume of simplex T_j yields the error estimator

$$\overline{\varepsilon}_j = \mathrm{vol}(T_j) \cdot \varepsilon_j = \mathrm{vol}(T_j)^{(p_j+1)/d+1}.$$

It only depends on the volume of simplex T_j and the theoretical order of convergence $\mathcal{O} = (p_j + 1)/d$. For an overall error estimator we sum again over all simplices

$$\overline{\varepsilon}_{\mathcal{O}} = \sum_{j=1}^{m} \overline{\varepsilon}_j.$$

4.2.3 Numerical Results for Test Functions

Smooth Functions

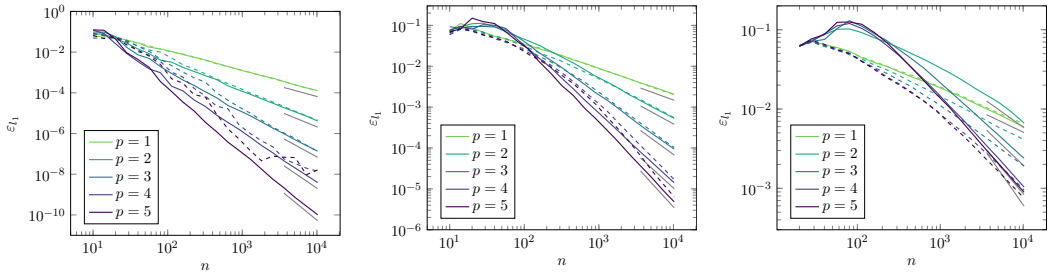
First we tested the simplex stochastic collocation algorithm with some smooth function $f \in C^\infty([0,1]^d)$

$$f(\mathbf{x}) = \prod_{i=1}^{d} \sin(\pi x^{(i)}),$$

for the Monte Carlo based error estimator $\hat{\varepsilon}_j$ with and without the local extremum conserving condition. In Figure 4.5 we see that the algorithm without the local extremum conserving condition yields slightly better results. Since the function is smooth, oscillations due to kinks or jumps cannot occur. Enforcing the local extremum conservation decreases the polynomial degree p_j if the function $f(\mathbf{x})$ itself has some small oscillations in simplex T_j. This reduction of the polynomial degree is not necessary and impairs convergence. But with increasing dimension we benefit from using the condition of local extremum conservation in the pre-asymptotic behavior. Therefore, we will use a weaker formulation of the local extremum conserving condition for dimensions $d \geq 4$ in the following. We will only reduce the polynomial degree of the approximation by one if it does not hold that

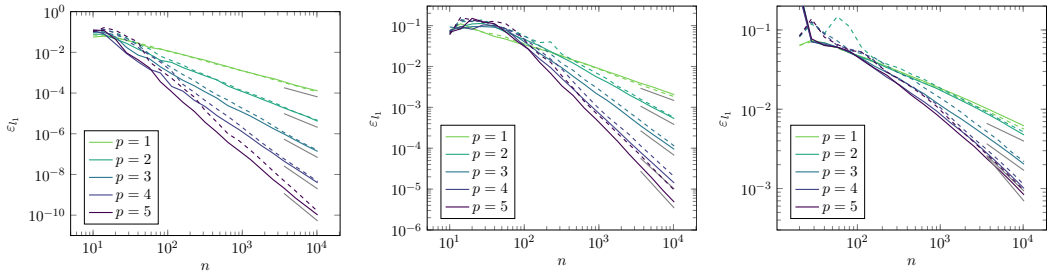
$$\min_{\mathbf{x}\in T_j} g_j(\mathbf{x}) + \delta \geq \min_{\mathbf{x}_i\in T_j} f(\mathbf{x}_i) \quad \wedge \quad \max_{\mathbf{x}\in T_j} g_j(\mathbf{x}) - \delta \leq \max_{\mathbf{x}_i\in T_j} f(\mathbf{x}_i)$$

with $\delta = 0.5(\max_{\mathbf{x}_i\in T_j} f(\mathbf{x}_i) - \min_{\mathbf{x}_i\in T_j} f(\mathbf{x}_i))$. This δ-local extremum conserving (δ-LEC) condition allows small oscillations in the approximation and improves the pre-asymptotic behavior without affecting the convergence. See Figure 4.6 (c) for the error in $d = 4$ dimensions with this weaker condition.



(a) $d = 2$ (b) $d = 3$ (c) $d = 4$

Figure 4.5: Shown is l_1 error evaluated at 10^6 random points versus the number n of sampling points for the smooth test function in $d = 2, 3, 4$ dimensions with (dashed lines) and without (solid lines) the LEC condition. The theoretical convergence rates are colored in gray.



(a) $d = 2$ (b) $d = 3$ (c) $d = 4$

Figure 4.6: Shown is the l_1 error evaluated at 10^6 random points versus the number n of sampling points for the smooth test function in $d = 2, 3, 4$ dimensions with (dashed lines) and without (solid lines) the kink information at $f(\mathbf{x}) = 0.7$. The theoretical convergence rates are colored in gray. Dimensions $d = 2$ and $d = 3$ are without the LEC condition and $d = 4$ is with the δ-LEC condition.

Above we stated that calculating two approximations on both sides of an assumed kink does not affect the convergence rate if in fact there is no kink. In order to verify this, we took the same test function $f(\mathbf{x}) = \prod_{i=1}^{d} \sin(\pi x_i)$ and assumed a kink at $f(\mathbf{x}) = 0.7$. See Figure 4.6 for the results. Assuming a kink yields slightly larger errors, but in all cases the desired convergence rates are attained. The difference between assuming and not assuming a kink decreases with increasing number n of sampling points. Note that for $d = 4$ dimensions we have already used the δ-local extremum conservation.

Non-Smooth Functions

Consider the test function

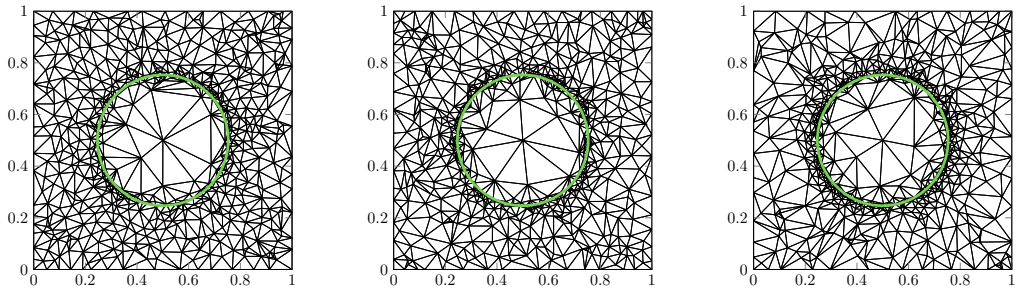
$$f(\mathbf{x}) = \min\left(\prod_{i=1}^{d} \sin(\pi x^{(i)}), 0.7\right).$$

To verify the convergence rates we calculate the interpolation error as l_1 norm be-

tween $f(\mathbf{x})$ and $g(\mathbf{x})$ evaluated at n_{MC} uniformly distributed Monte Carlo points:

$$\varepsilon_{l_1} = \sum_{i=1}^{n_{\mathrm{MC}}} \frac{|f(\mathbf{x}_{\mathrm{MC},i}) - g(\mathbf{x}_{\mathrm{MC},i})|}{n_{\mathrm{MC}}}. \tag{4.4}$$

First we show numerical results for the original simplex stochastic collocation version [WI12b] with the local extremum conservation and no special approximation for kinks. As expected, enforcing the local extremum conservation reduces the polynomial degree near the kink which results in a larger error estimator and thus in a finer discretization, see Figure 4.7. The higher the polynomial degree is, the more points are added near the kink. We expected this behavior because the smooth part of f can be better approximated with polynomials of higher degree, whereas increasing the degree of the interpolating polynomials does not benefit approximating the kink.

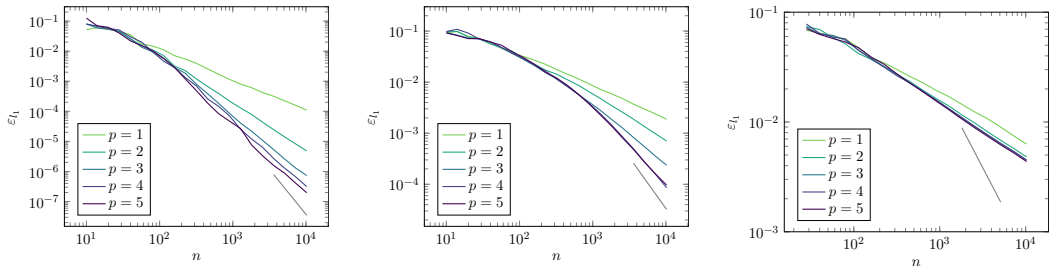


(a) $p_j = 2$ (b) $p_j = 3$ (c) $p_j = 4$

Figure 4.7: Original SSC: Shown is an adaptively refined Delaunay triangulation of $n = 640$ sampling points for the non-smooth test function with different polynomial degrees of $p_j = 2, 3, 4$ and the l_1 error estimator $\tilde{\varepsilon}_j$. The location of the kink is marked in green.

See Figure 4.8 for the convergence rates of the original simplex stochastic collocation. In $d = 2$ dimensions, the desired convergence rates are attained for small polynomial degrees $p = 1, 2, 3$. Increasing the polynomial degrees up to $p = 4$ and $p = 5$ improves the convergence rate slightly to -2.1, but the theoretical rates of -2.5 and -3 are not attained. In $d = 3$ dimensions the errors for $p = 4$ and $p = 5$ are already the same with a maximal rate of -1.5 instead of -2. For dimensions larger or equal to $d = 4$ using polynomials of higher degree is not beneficial and the maximally attained convergence rate is -0.5. Therefore, the original simplex stochastic collocation is useless for computing statistics of the solution in $d \geq 4$ dimensions. For these cases Monte Carlo methods provide better results with less computational effort.

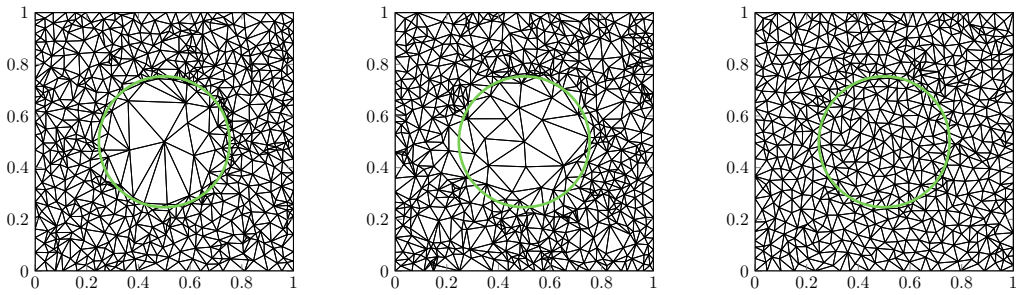
Now we analyze the modified simplex stochastic collocation method. By checking if the function value $f(\mathbf{x}_i)$ is smaller or equal to 0.7 we can assign each sampling point x_i either to $\Omega_1 = \{\mathbf{x} \in \Omega : f(\mathbf{x}) < 0.7\}$ or to $\Omega_2 = \{\mathbf{x} \in \Omega : f(\mathbf{x}) = 0.7\}$. An adaptively refined Delaunay triangulation with the l_1 error estimator $\hat{\varepsilon}_j$ for $p_j = 5$ and $n = 640$ sampling points can be found in Figure 4.9(a). As expected, the sampling



(a) $d = 2$ (b) $d = 3$ (c) $d = 4$

Figure 4.8: Original SSC: Shown is the l_1 error evaluated at 10^6 random points versus the number n of sampling points for different dimensions $d = 2, 3, 4$ with the l_1 error estimator $\tilde{\varepsilon}_j$. The desired convergence rates for a polynomial degree of $p = 5$ are plotted in gray.

points are more or less uniformly distributed over the parameter space Ω where the function value is not constant. In the center of our domain where the function value is constant, the areas of the triangles are significantly larger. The triangulation in Figure 4.9(b) for the root mean square error estimator $\tilde{\varepsilon}_j$ looks quite similar: there are fewer triangles in the center than around it where the triangles are less uniformly sized as for the estimator $\hat{\varepsilon}_j$. In contrast, the resulting triangulation for the function-independent error estimator $\overline{\varepsilon}_j$ is uniform and it is not possible to recognize the location of the kink, see Figure 4.9(c).

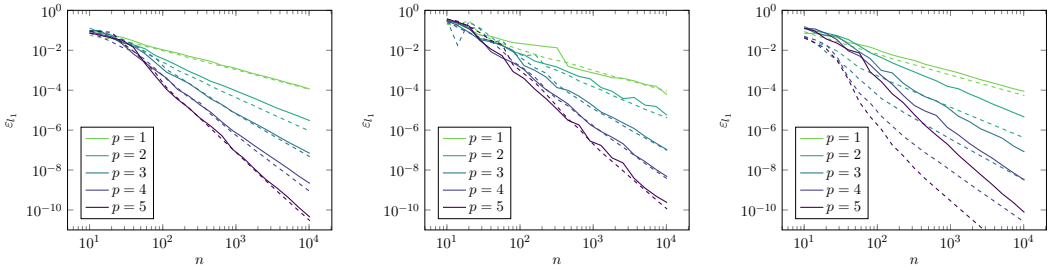


(a) Error estimator $\tilde{\varepsilon}_j$ (b) Error estimator $\hat{\varepsilon}_j$ (c) Error estimator $\overline{\varepsilon}_j$

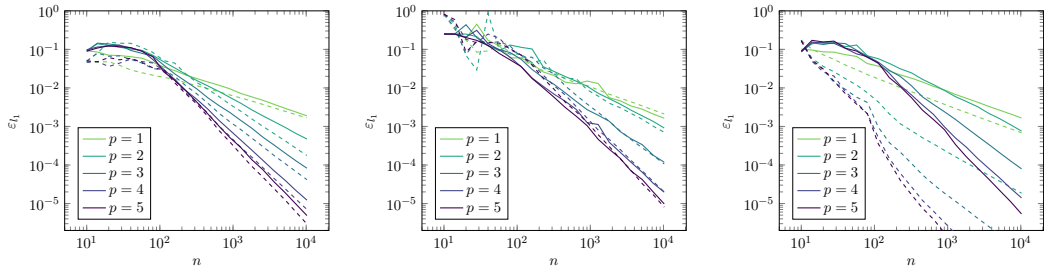
Figure 4.9: Modified SSC: Shown is an adaptively refined Delaunay triangulation with $n = 640$ sampling points for the function $f(\mathbf{x}) = \min(\prod_{i=1}^{d} \sin(\pi x^{(i)}), 0.7)$ in 2d with a polynomial degree of $p_j = 5$ for the l_1 error estimator $\tilde{\varepsilon}_j$ (a), for the root mean square error estimator $\hat{\varepsilon}_j$ (b), and for the function-independent error estimator $\overline{\varepsilon}_j$ (c). The location of the kink is marked in green.

In all shown dimensions $d = 2, 3, 4$ nearly all theoretical convergence rates of ε_{l_1} evaluated at $n_{\mathrm{MC}} = 10^6$ Monte Carlo points are attained for the l_1 error estimator as well as for the root mean square error estimator $\hat{\varepsilon}_j$ and the error estimator $\overline{\varepsilon}_j$, cf. Figure 4.10. Only in four dimensions increasing the polynomial degree from four to five does not improve the convergence rate. The l_1 error estimator yields the best results and the smoothest convergence. The total errors reached with error estimator

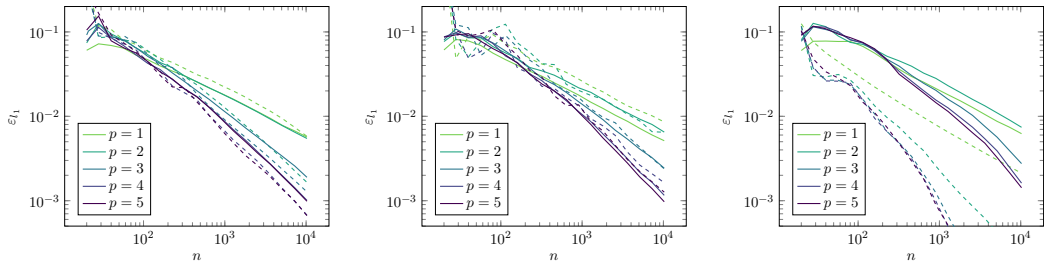
$\overline{\varepsilon}_j$ look quite similar, but as expected the estimated overall error differs greatly from the real error because it is not solution-based. The pointwise error estimator $\tilde{\varepsilon}_j$ yields the worst results with a convergence not as smooth as with the other ones. But in two and three dimensions the estimated overall error is close to the real error. Comparing these total errors with those obtained with the original simplex stochastic collocation method, shows that the modification yields significantly better results. The total error for the maximal number of points was improved from 10^{-7} to 10^{-10} in two dimensions, from 10^{-4} to 10^{-6} in three dimensions, and from $5 \cdot 10^{-3}$ to 10^{-3} in four dimensions.



(a) l_1 error for $\tilde{\varepsilon}_j$, 2d (b) l_1 error for $\hat{\varepsilon}_j$, 2d (c) l_1 error for $\overline{\varepsilon}_j$, 2d



(d) l_1 error for $\tilde{\varepsilon}_j$, 3d (e) l_1 error for $\hat{\varepsilon}_j$, 3d (f) l_1 error for $\overline{\varepsilon}_j$, 3d

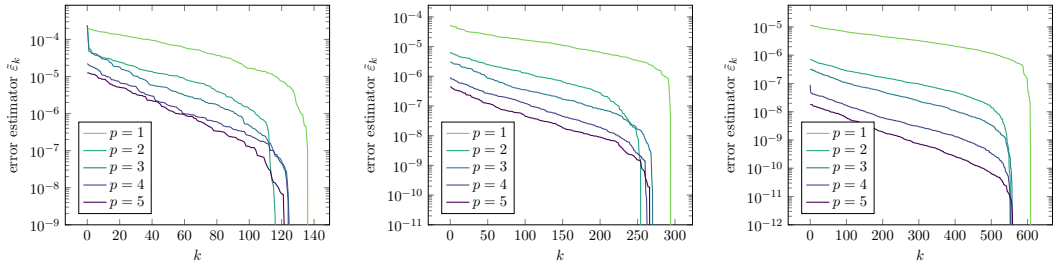


(g) l_1 error for $\tilde{\varepsilon}_j$, 4d (h) l_1 error for $\hat{\varepsilon}_j$, 4d (i) l_1 error for $\overline{\varepsilon}_j$, 4d

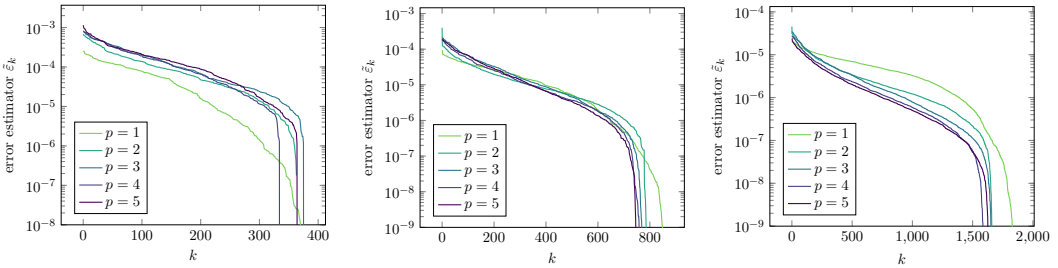
Figure 4.10: Modified SSC: Shown is the l_1 error evaluated at 10^6 random points (solid) and the error estimator (dashed) versus the number n of interpolation points for the l_1 error estimator $\tilde{\varepsilon}_j$ (a), the root mean square error estimator $\hat{\varepsilon}_j$ (b), and the error estimator $\overline{\varepsilon}_j$ (c).

4.2.4 Multiple Refinements

In order to parallelize the refinement, at each step the $m_{\text{ref}} \geq 1$ simplices with the largest error estimator can be refined. Thus, the function evaluations for the new sampling points can be done simultaneously and the expensive update of the Delaunay triangulation needs just to be done only once instead of m_{ref} times. Figure 4.11 shows the l_1 error estimator versus the indices of simplices. Independent of the dimension, the polynomial degree, and the number of interpolation points, the error estimator slowly decreases over most simplices. Only for a small percentage of simplices the error estimator is significantly smaller than for the rest. Thereby it is reasonable to add several sampling points at once.



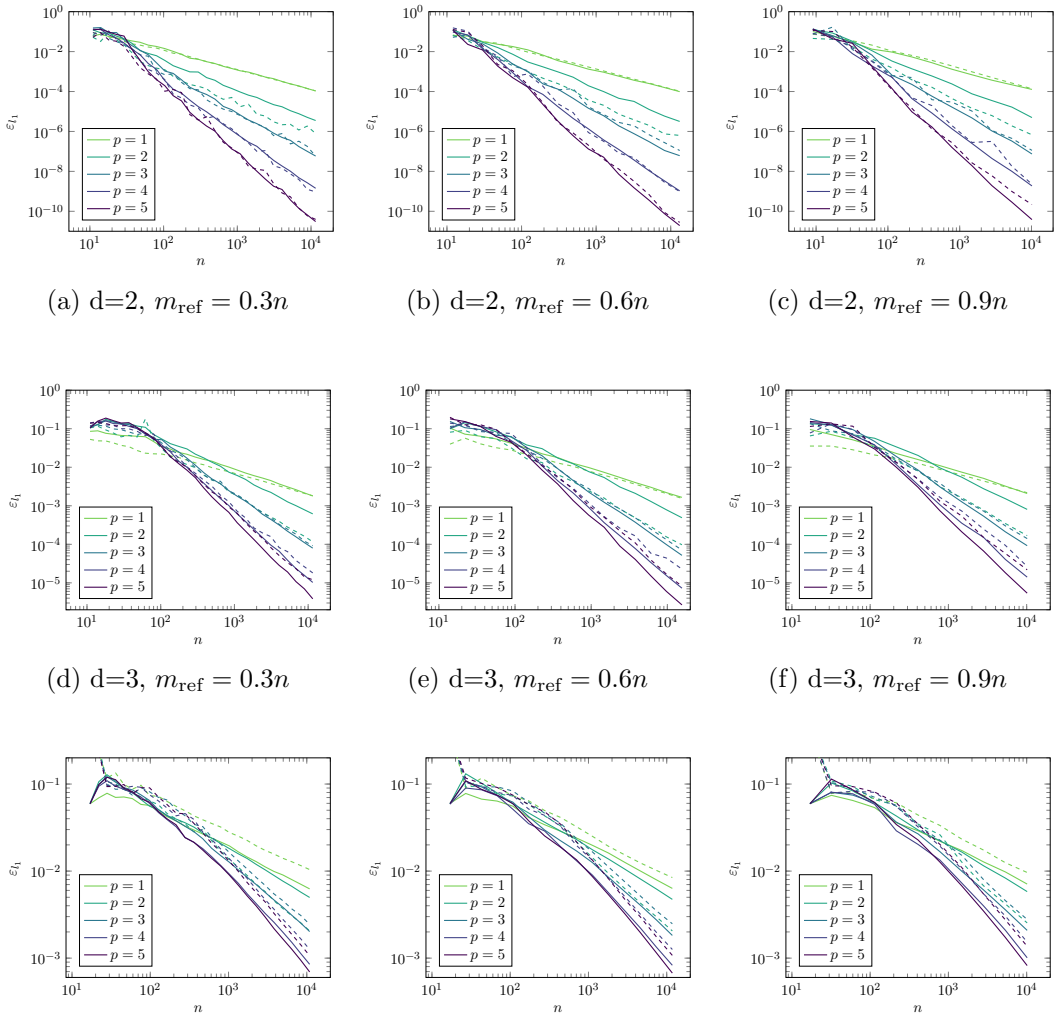
(a) d=2, m=80 (b) d=2, m=160 (c) d=2, m=320



(d) d=3, m=80 (e) d=3, m=160 (f) d=3, m=320

Figure 4.11: Shown is the distribution of the l_1 error estimator over the indices of the sorted simplices.

Figure 4.12 shows the convergence rates for multiple refinements where we used the Monte Carlo based error estimator $\tilde{\varepsilon}_j$ and at each step added $0.3n$, $0.6n$, or $0.9n$ points, respectively, to the current discretization consisting of n sampling points. Since the number of newly added sampling points does not influence the convergence, it is reasonable to refine multiple simplices to save computational time.



(a) d=2, $m_{\text{ref}} = 0.3n$ (b) d=2, $m_{\text{ref}} = 0.6n$ (c) d=2, $m_{\text{ref}} = 0.9n$

(d) d=3, $m_{\text{ref}} = 0.3n$ (e) d=3, $m_{\text{ref}} = 0.6n$ (f) d=3, $m_{\text{ref}} = 0.9n$

(g) d=4, $m_{\text{ref}} = 0.3n$ (h) d=4, $m_{\text{ref}} = 0.6n$ (i) d=4, $m_{\text{ref}} = 0.9n$

Figure 4.12: Shown is the l_1 error evaluated at 10^6 random points (solid) and the error estimator (dashed) versus interpolation points n for l_1 error estimator $\tilde{\varepsilon}_j$. At each refinement step 30% (left column), 60% (central column), and 90% (right column) of the old sampling points were added.

4.3 Comparison with VPS Models

Since Voronoi piecewise surrogate (VPS) models [RSP+17] are very similar to simplex stochastic collocation, we compare both methods. In Voronoi piecewise surrogate models the parameter space is discretized by Voronoi cells, the dual graph to a Delaunay triangulation. Similar to simplex stochastic collocation, the piecewise polynomial approximation of degree p is computed by polynomial regression of the $2(p+d)!/(p!d!)$ nearest neighbors of each cell. The difference between the function values of the nearest neighbors and the function value of the cell's center is not allowed to exceed a user-defined threshold, otherwise a discontinuity is detected. Because of a similar type of approximation we expect similar convergence results. We tested our modified version of the simplex stochastic collocation with the same functions as in [RSP+17]:

$$f_1(\mathbf{x}) = -\prod_{k=1}^{d} \exp(-(x_k-1)^2) + \exp(-0.8(x_k+1)^2)$$

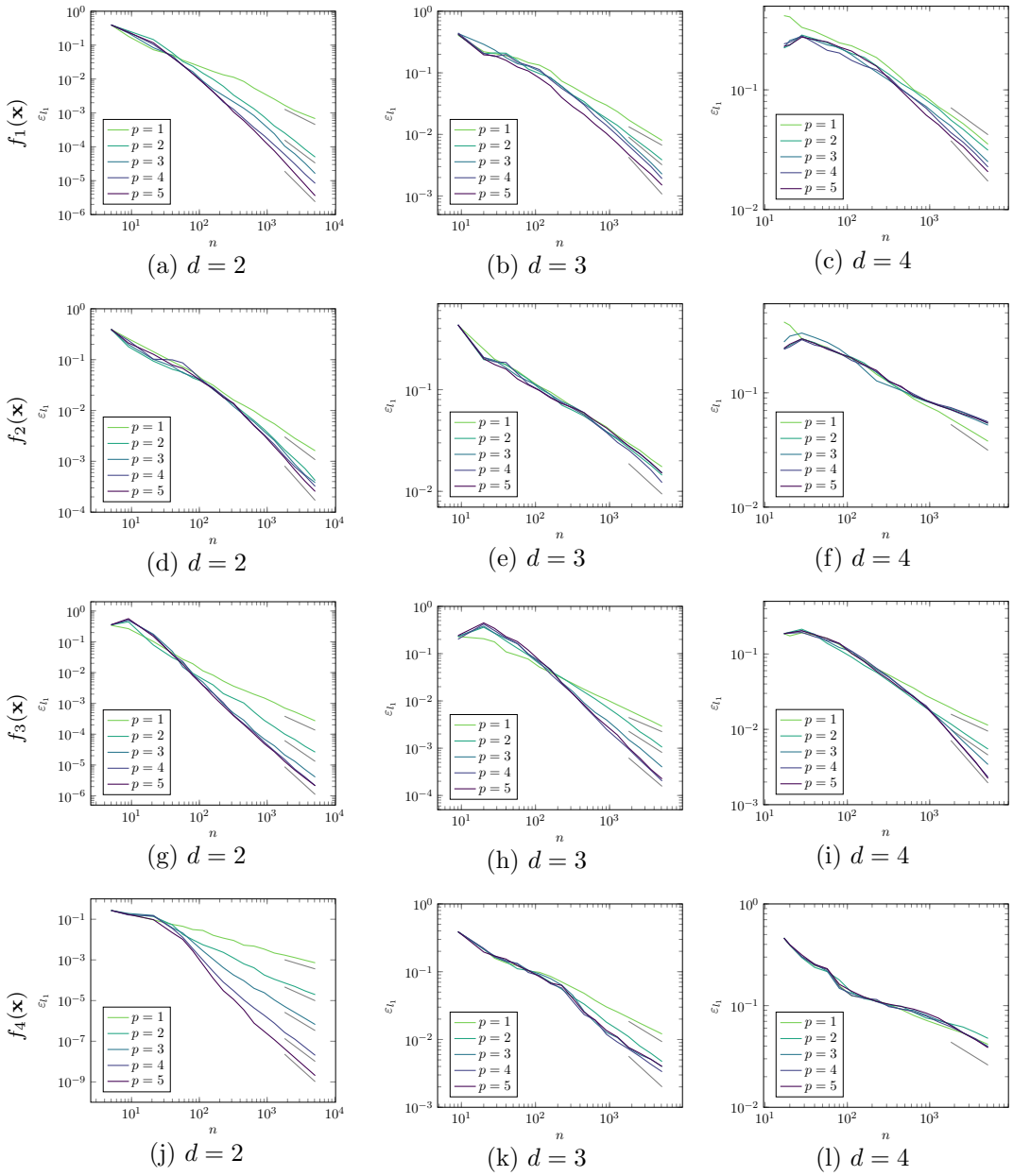
$$f_2(\mathbf{x}) = -\prod_{k=1}^{d} \exp(-(x_k-1)^2) + \exp(-0.8(x_k+1)^2) - 0.05\sin(8(x_k+0.15))$$

$$f_3(\mathbf{x}) = \left(\sum_{k=1}^{d} x_k^2\right)^{1/d}$$

$$f_4(\mathbf{x}) = \left(\prod_{k=1}^{d} \frac{1+\cos(2\pi x_k)}{2}\right)^{1/d}$$

over their standard test domains $[-2,2]^d$, $[-2,2]^d$, $[-1,1]^d$, and $[0,1]^d$, respectively. These functions have different features, making them challenging for approximation. The first function is the smoothed Herbie function, which is a relatively smooth function. The second function, the Herbie function, has a high frequency sine component that creates a large number of local minima and maxima. The third one, the circular cone, has a single singularity at the origin and its local Lipschitz constant is unity everywhere. The fourth one is the planar cross, which expands the cone's single singularity along the main axes. Since the first two functions are globally smooth and function three has no kink according to our definition, we used in theses cases a simplex stochastic collocation without any special kink handling. The fourth function is approximated separately is each smooth region Ω_k and for simplices containing the kink we take take the minimum of all approximations. See Figure 4.13 for the convergence results with our modified simplex stochastic collocation. We used multiple refinements that doubled the number of points after two refinement steps. For the smooth Herbie function f_1 all errors are of the same order as for the Voronoi piecewise surrogate models, cf. [RSP+17]. Increasing the polynomial degree results in better convergence rates but, nevertheless, the theoretical convergence rates of $\mathcal{O}(-(p+1)/d)$ cannot be obtained for $p \geq 3$. For the Herbie function f_2, where increasing the polynomial degree is not beneficial, our method yields worse results, whereas it yields better results for the cone f_3. With our method, the explicit kink approximation for the cross f_4 significantly improves convergence in $d = 2$ dimen-

sions, where all theoretical convergence rates are attained. In $d = 3$ dimensions the benefit of the kink approximation is still visible, in $d = 4$ dimensions it is no longer visible due to the poor approximability of the function's smooth parts. In error estimate (4.2) the partial derivatives of f_4 and the number of terms in the sum increase with increasing polynomial degree. Concluding, we can say that both methods suffer from the same challenging features, but the explicit kink approximation is profitable if the rest of the function is easy to approximate.



(a) $d = 2$ (b) $d = 3$ (c) $d = 4$

(d) $d = 2$ (e) $d = 3$ (f) $d = 4$

(g) $d = 2$ (h) $d = 3$ (i) $d = 4$

(j) $d = 2$ (k) $d = 3$ (l) $d = 4$

Figure 4.13: The l_1 error evaluated at 10^6 random points versus the number n of sampling points for the test functions f_1, f_2, f_3 and f_4 in $d = 2, 3, 4$ dimensions. The theoretical convergence rates are colored in gray.

4.4 Statistics of the Approximated Function

When simulating gas networks some input data can be uncertain like the pressure of the injected gas at input nodes or the flux of the extracted gas at demand nodes. The response of the gas network to these uncertainties are expressed by the pressure and temperature at nodes and the flux through pipes. We are interested in statistics of these physical quantities like the expected value, variance, median, or cumulative density function (cdf).

4.4.1 The Expectation and Variance

The expectation of a function $f(\mathbf{x})$ of a random variable $\mathbf{x} \in \Omega$ with the density function $\rho(\mathbf{x})$ is defined as

$$\mathbb{E}[f] = \int_{\Omega} f(\mathbf{x})\rho(\mathbf{x})\,\mathrm{d}\mathbf{x}.$$

Using the approximations $g_j(\mathbf{x})$ on the simplices T_j we can approximate $\mathbb{E}[f]$ by evaluating a quadrature rule Q for the approximation $\tilde{f}$ of f, i.e.

$$\mathbb{E}[f] \approx Q(\tilde{f}) = \sum_{j=1}^{m} Q(g_j).$$

The quadrature rule Q can either be a Monte Carlo integration or a Gaussian quadrature. Notice that we do not need to evaluate f for any quadrature point but only the approximations g_j.

The variance of a function $f(\mathbf{x})$ of a random variable $\mathbf{x} \in \Omega$ with the density function $\rho(\mathbf{x})$ is defined as the squared distance of the function from its mean

$$\mathbb{V}[f] = \int_{\Omega} \left(f(\mathbf{x}) - \mathbb{E}[f]\right)^2 \rho(\mathbf{x})\mathrm{d}\mathbf{x}$$

and can be written as

$$\mathbb{V}[f] = \mathbb{E}[f^2] - \mathbb{E}[f]^2.$$

We approximate the variance in the same way as the expectation, namely by

$$\mathbb{V}[f] \approx Q(\tilde{f}^2) - Q(\tilde{f})^2$$

using again Monte Carlo integration or Gaussian quadrature to calculate the integrals.

The Error Estimation

The absolute error of the expectation $|\mathbb{E}[f] - Q(\tilde{f})|$ can be estimated as

$$\begin{aligned}\left|\mathbb{E}[f] - Q(\tilde{f})\right| &\leq \left|\mathbb{E}[f] - \mathbb{E}[\tilde{f}]\right| + \left|\mathbb{E}[\tilde{f}] - Q(\tilde{f})\right| \\ &\leq \int_\Omega \left|f(\mathbf{x}) - \tilde{f}(\mathbf{x})\right| \rho(\mathbf{x})\, \mathrm{d}\mathbf{x} + \left|\mathbb{E}[\tilde{f}] - Q(\tilde{f})\right| \\ &\leq \int_\Omega \underbrace{\left|f(\mathbf{x}) - \tilde{f}(\mathbf{x})\right|_\infty}_{\varepsilon_I(f)} \rho(\mathbf{x})\, \mathrm{d}\mathbf{x} + \underbrace{\left|\mathbb{E}[\tilde{f}] - Q(\tilde{f})\right|}_{\varepsilon_Q(f)} \\ &= \varepsilon_I(f) + \varepsilon_Q(f).\end{aligned}$$

The interpolation error $\varepsilon_I(f)$ can be estimated by (4.2). If we choose the quadrature formula such that the quadrature error $\varepsilon_Q(f)$ is at most of the same order of magnitude as the interpolation error $\varepsilon_I(f)$, then the approximation $Q(\tilde{f})$ of the expected value $\mathbb{E}[f]$ converges also with a rate of $\mathcal{O}(-(p+1)/d)$, provided that the partial derivatives are bounded.

The same rate can be obtained for the variance if the function and all partial derivatives are bounded. Using the triangle inequality we get the following two terms:

$$\left|\mathbb{V}[f] - (Q(\tilde{f}^2) - Q(\tilde{f})^2)\right| \leq \left|\mathbb{E}[f^2] - Q(\tilde{f}^2)\right| + \left|\mathbb{E}[f]^2 - Q(\tilde{f})^2\right| \tag{4.5}$$

Analogously to the expectation, the first term can be estimated by

$$\left|\mathbb{E}[f^2] - Q(\tilde{f}^2)\right| \leq \varepsilon_I(f^2) + \varepsilon_Q(f^2).$$

With $|f^2 - \tilde{f}^2| \leq |f - \tilde{f}|\,|f + \tilde{f}| \leq |f - \tilde{f}|\,(|f| + |\tilde{f}|)$ we obtain

$$\left|\mathbb{E}[f^2] - Q(\tilde{f}^2)\right| \leq (|f| + |\tilde{f}|)\varepsilon_I(f) + \varepsilon_Q(f^2).$$

Next we consider the second term of (4.5):

$$\begin{aligned}\left|\mathbb{E}[f]^2 - Q(\tilde{f})^2\right| &\leq \left|\mathbb{E}[f] - Q(\tilde{f})\right| \left(|\mathbb{E}[f]| + |Q(\tilde{f})|\right) \\ &\leq (\varepsilon_I(f) + \varepsilon_Q(f)) \left(|\mathbb{E}[f]| + |Q(\tilde{f})|\right).\end{aligned}$$

Assuming bounded $f, \tilde{f}, \mathbb{E}[f], Q(\tilde{f}) \leq C$, we get the following result for the error of the variance

$$\begin{aligned}\left|\mathbb{V}[f] - (Q(\tilde{f}^2) - Q(\tilde{f})^2)\right| &\leq 2C\, \varepsilon_I(f) + \varepsilon_Q(f^2) + 2C\, \varepsilon_I(f) + 2C\, \varepsilon_Q(f) \\ &\leq 4C\, \varepsilon_I(f) + 2C\, \varepsilon_Q(f) + \varepsilon_Q(f^2).\end{aligned}$$

Hence, by choosing the quadrature formula such that the quadrature errors $\varepsilon_Q(f)$ and $\varepsilon_Q(f^2)$ are at most of the same order of magnitude as the interpolation error $\varepsilon_I(f)$, yields again a convergence rate of $\mathcal{O}(-(p+1)/d)$.

4.4.2 The CDF and Median

For approximating the cumulative density function $\mathbb{P}[f(\mathbf{x}) \leq y]$, we discretize the value range of the approximation g with equidistant nodes $y_0, y_1, \ldots, y_n$ where $y_i = \min(f) + ih$ and $h = (\max(g) - \min(g))/n$. For each node y_i we determine the maximal domain $\Omega_i \subseteq \Omega$ such that $g(\mathbf{x}) \leq y_i$ for all $\mathbf{x} \in \Omega_i$. With the probabilities of these domains we obtain the function values of the cumulative density function because it holds

$$\mathbb{P}[g(\mathbf{x}) \leq y_i] = \mathbb{P}[\Omega_i].$$

As a last step we interpolate the cumulative density function between the nodes, e.g. with piecewise linear polynomials. Note that the interpolation must be monotonically increasing because otherwise the resulting function does not fulfill the requirements of a cumulative density function.

The median m of a function $f(\mathbf{x})$ of a random variable $\mathbf{x} \in \Omega$ is defined as

$$\mathbb{P}[f(\mathbf{x}) \leq m] = \mathbb{P}[f(\mathbf{x}) \geq m] = \frac{1}{2}.$$

It can either be approximated by inverting the cumulative density function for $\mathbb{P}[f(\mathbf{x}) \leq y] = 0.5$, see Figure 4.14, or by using the sample median $\tilde{m}$ of n randomly drawn Monte Carlo points $\mathbf{x}$. Let n be odd and $\mathbf{x}_i$ be sorted such that $f(\mathbf{x}_i) \leq f(\mathbf{x}_{i+1})$ for all $i = 1, \ldots, n-1$, then the sample median $\tilde{m}$ is defined as

$$\tilde{m} = f(\mathbf{x}_{\lceil n/2 \rceil}).$$

If n is even, then the median $\tilde{m}$ is defined as

$$\tilde{m} = \frac{f(\mathbf{x}_{n/2}) + f(\mathbf{x}_{n/2+1})}{2}.$$

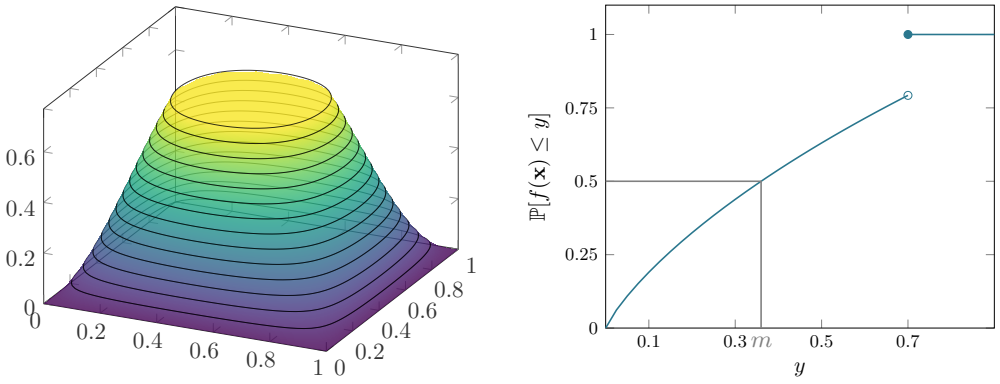


Figure 4.14: Surface plot of the test function $f(\mathbf{x})$ (left) and corresponding cdf for uniformly distributed $\mathbf{x}$ (right). The median m is marked in gray.

5 Numerical Results for Gas Networks

In this chapter we apply our new version of simplex stochastic collocation to a real gas network. The network has one supply node, 37 demand nodes, several pipes, and five control valves which reduce the high pressure of about 27 bar at the supply node stepwise to pressures of around 16, 8, and 4 bar at the demand nodes. See Figure 5.1 for a schematic drawing of the network. Different pressure levels are colored in different colors. In all tests the quantity of interest is the outgoing pressure $f(\mathbf{x})$ at the right control valve. Depending on the uncertain parameters of outgoing pressure x_1 at the left valve, and the amount of withdrawn gas at demand nodes x_2, x_3, and x_4, the right valve is in an active or bypass mode which is checked by comparing the outgoing pressure with the preset pressure. The lower the outgoing pressure x_1 is, and the higher the withdrawn amount of gas is, the lower is the incoming pressure $f(\mathbf{x})$ at the right control valve.

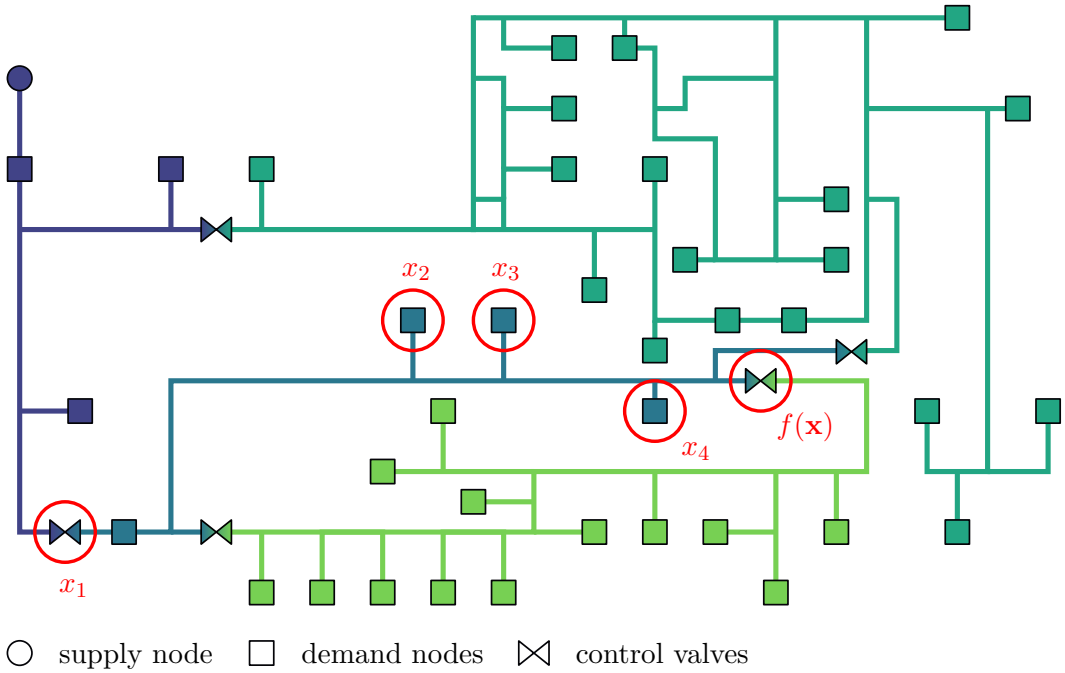


Figure 5.1: Test gas network with one supply node, 37 demand nodes, and five pressure control valves.

5.1 The Model Errors

The gas flow model suffers from different errors. As an example, we will discuss two types of errors. In practice, complete terms of the equation of momentum conservation (2.4) are often omitted. In [Osi84] the contributions

$$\delta_1 = \frac{\int_0^L \partial_t q \,\mathrm{d}x}{p(0,t) - p(L,t)} \cdot 100\%,$$
$$\delta_2 = \frac{\int_0^L \frac{\lambda v^2}{2D} \rho \,\mathrm{d}x}{p(0,t) - p(L,t)} \cdot 100\%,$$
$$\text{and} \quad \delta_3 = \frac{(\rho v^2)_{x=L} - (\rho v^2)_{x=0}}{p(0,t) - p(L,t)} \cdot 100\%$$

are computed for three different settings. In all three settings δ_2 has the largest value of $150\% - 170\%$, and δ_1 and δ_3 are less than 1%. Hence, neglecting the terms $\partial_t q$ and $\partial_x(\rho v^2)$ results in an error of approximately 1% which is sufficiently accurate for many applications.

Our gas solver takes these terms into account, but in the derivation of the law of momentum conservation (Subsection 2.1.2) we assumed uniform velocity, pressure, and density over the pipe but in reality all these variables slightly vary. To get an idea of which scale these errors are, we split the longest pipe in the test network with a length of approximately 10 km into s equally sized sub-pipes and check how much the pressure $f(\mathbf{x})$ varies for a fixed $\mathbf{x}$. See Figure 5.2 for the error in the pressure $f(\mathbf{x})$ which arises from splitting the longest pipe into s parts. The reference pressure was calculated by splitting the pipe into 128 parts. Hence, the error of using only one pipe for the simulation is of order 10^{-4}. Of course, this is only the error caused by one pipe and all other pipes induce further errors, but at least we have a lower bound for the overall error. Therefore, it is sufficient for the simplex stochastic collocation to reach an error of the same order of magnitude.

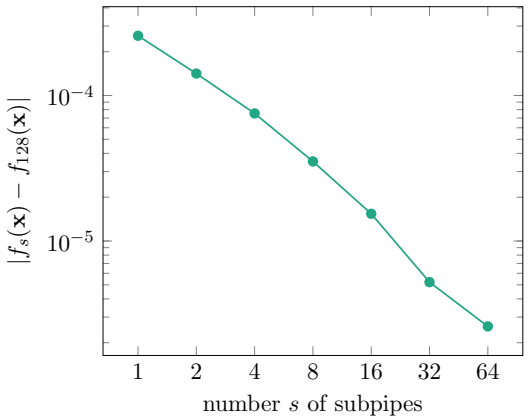
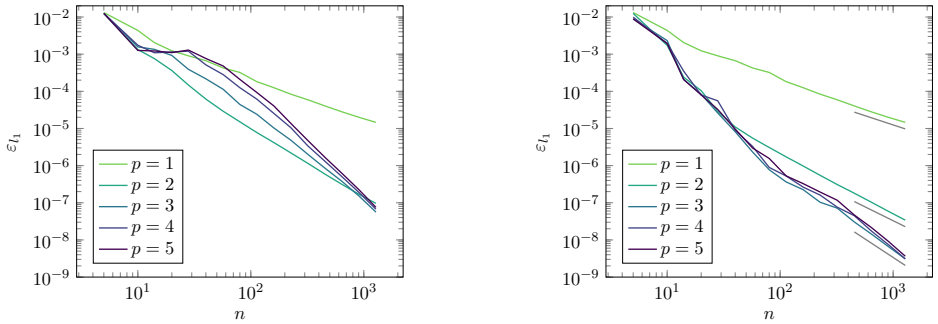


Figure 5.2: The absolute error in the pressure $f(\mathbf{x})$ caused by assuming uniform gas properties along the pipe.

5.2 Input Uncertainties in Two Dimensions

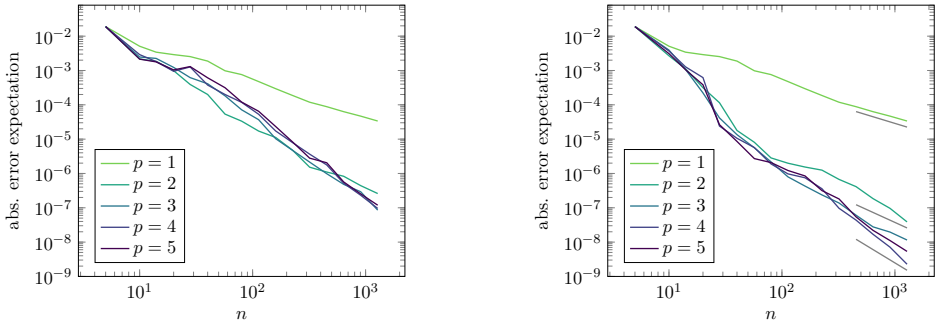
5.2.1 Function Approximation and Expected Value

First, we vary the outgoing pressure x_1 of the left control valve uniformly between 8.5 bar and 9.5 bar, and the demanded power x_2 uniformly between 160 MW and 200 MW. The remaining powers are fixed, in particular $x_3 = 250$ MW and $x_4 = 17$ MW. See Figure 5.3 for the comparison of the original simplex stochastic collocation (a) with the new simplex stochastic collocation (b). The new version yields better results than the original one, the smallest error reached is 10^{-2} times smaller. In the original version it makes no difference whether polynomials of degree $p = 2$ or higher are used. In the new version the desired convergence rates (marked in gray) for $p = 1$, $p = 2$, and $p = 3$ are obtained. Increasing the polynomial degree to $p = 4$ or $p = 5$ yields no improvement in the rate. Similar results are valid for the expected value, where a reference value was computed with a polynomial degree of $p = 5$ and $m = 5120$ sampling points, see Figure 5.4. The original simplex stochastic collocation needs $m \approx 50$ sampling points to achieve an accuracy of 10^{-4}, whereas the new version only needs $m \approx 30$ sampling points.



(a) Original SSC (b) New SSC

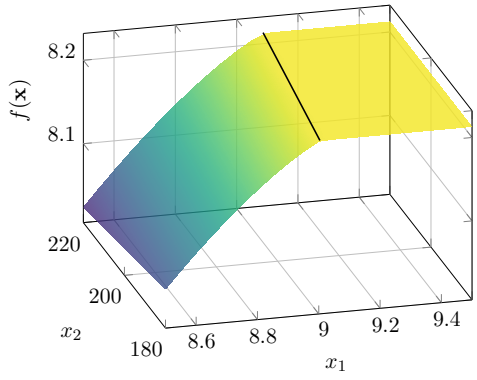
Figure 5.3: $d = 2$. The l_1 error evaluated at 10^6 random points versus the number n of interpolation points with l_1 error estimator $\tilde{\varepsilon}_j$ for the original SSC without kink information (a), and for the new version with kink information (b).



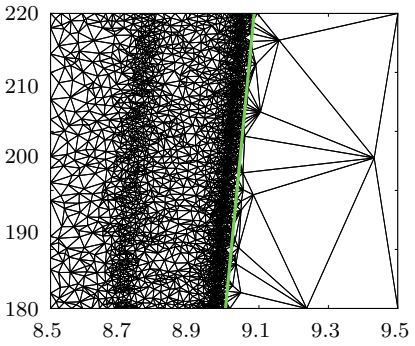
(a) Original SSC (b) New SSC

Figure 5.4: $d = 2$. The absolute error in the expected value versus the number n of interpolation points with l_1 error estimator $\tilde{\varepsilon}_j$ for the original SSC without kink information (a), and for the new version with kink information (b).

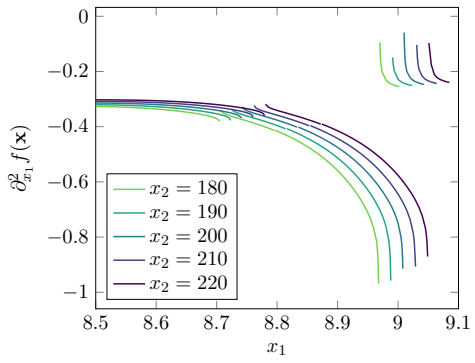
The question is why we cannot improve the convergence rate using higher order polynomials as it was the case for the synthetic test function in the previous section? To answer this question see Figure 5.5. The surface plot of the function (a) is inconspicuous. But if we look at the resulting triangulation (b), we can see that there are two areas in which the error estimator places an unexpected number of points. This indicates discontinuities. Because of this, we approximated the second order partial derivative $\partial_{x_1} f(\mathbf{x})$ with the second order finite difference quotient (c) and, hence, we can see two jumps in the second partial derivative which explain the poor convergence results. These jumps are not caused by the physical properties of gas flow but by numerical issues of the solver. For the developers of the solver it is not surprising that such jumps arise. Since it is not predictable where they arise, we do not have a possibility to adapt our method. Theoretically, these jumps could be avoided, but this would effect the solution process at other points and convergence to a solution would not any longer be guaranteed. Moreover, this version of the solver is completely sufficient for industrial applications. Therefore, there is no reason to improve it at this point, and we must use it as it is.



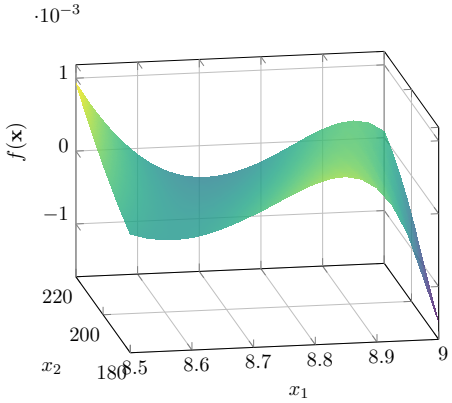
(a) Surface plot



(b) Triangulation



(c) Second derivative



(d) Difference to quadratic regression

Figure 5.5: Function $f(\mathbf{x})$ resulting from gas network simulation.

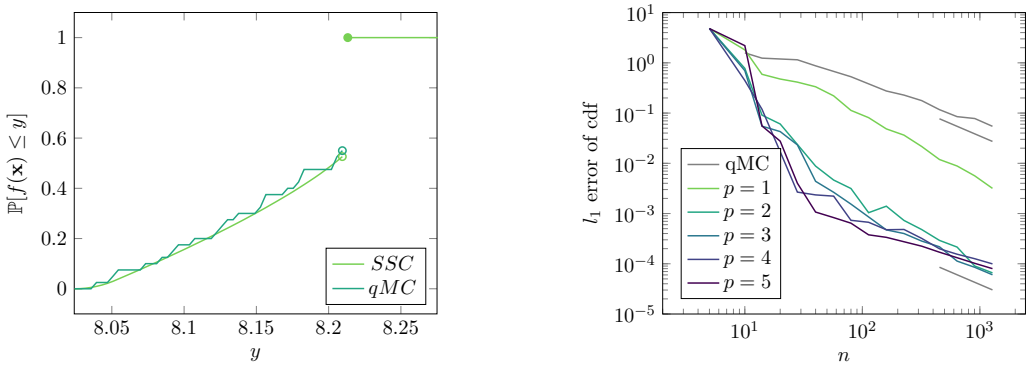
Because the original version does not have any information about the kink in the function $f(\mathbf{x})$, the convergence rates for polynomial degrees of $p \geq 2$ are the same. The new version has some information about the kink in the function, but no information about the kink in the first derivative (corresponding to the jump in the second derivative) and, therefore, the convergence rates for $p \geq 3$ are the same. At first glance, we only improved the convergence rate from -1.5 to -2, but at second glance, we see that our new simplex stochastic collocation has a significantly better pre-asymptotic behavior. This is due to the fact that the linear and quadratic terms of $f(\mathbf{x})$ contribute most, whereas the higher order terms are only of magnitude 10^{-3}. See Figure 5.5 (d) for the difference between the function $f(\mathbf{x})$ and a quadratic regression at the left side of the kink.

5.2.2 The Cumulative Density Function

Now we calculate the cumulative density function. The cumulative density function is more important for analyzing the gas network than the expected value because we can use it to predict in what percentage of cases a certain pressure or flux value is exceeded or undershot. As in Section 4.4.2 described, we discretize the value range

$$I = \left[\min_{\mathbf{x}\in\Omega} f(\mathbf{x}),\ \max_{\mathbf{x}\in\Omega} f(\mathbf{x})\right]$$

with 50 equidistant nodes y_i and calculate $\mathbb{P}[f(\mathbf{x}) \leq y_i]$ by evaluating our approximation, and not the function itself, at 10^{12} Monte Carlo sampling points. Between the nodes y_i, the cumulative density function is approximated by a piecewise linear interpolation. See Figure 5.6(a) for a plot of the cumulative density function computed by using the simplex stochastic collocation with only $m = 40$ evaluation points. For the polynomial degrees $p = 2, \ldots, 5$ we get a result which is optically identical to the reference solution computed with the stochastic simplex collocation and $m = 5120$ sampling points.



(a) CDF (b) Convergence plot

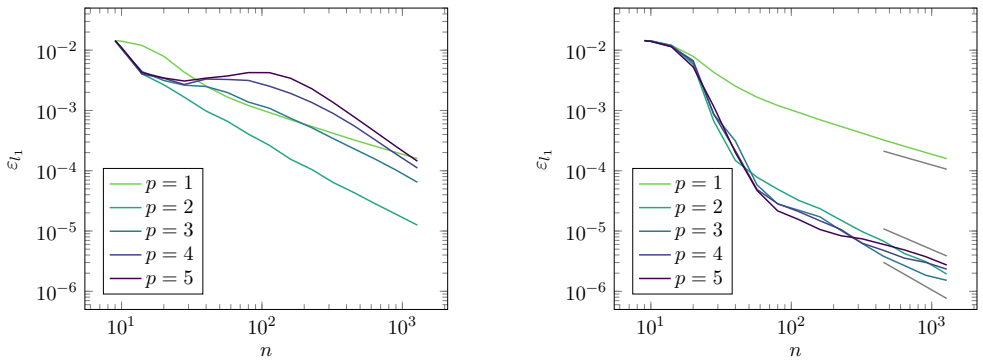
Figure 5.6: Shown is a plot of the cumulative density function in $d = 2$ dimensions for the SSC and qMC sampling points (a), and the corresponding convergence rates (b). Both approximations of the cumulative density function are computed by evaluating the function at $m = 40$ points.

In contrast, the cumulative density function computed with $m = 40$ Halton points looks significantly worse. In Figure 5.6(b) the corresponding convergence rates are plotted. As in the case of the expected value, there is no benefit of using higher order polynomials but, again, the pre-asymptotic is very good and the obtained errors are significantly smaller than for the Halton sequence. Here, the attained convergence rate is only $\mathcal{O}(-1)$, whereas it was $\mathcal{O}(-2)$ for the expectation.

5.3 Input Uncertainties in Three Dimensions

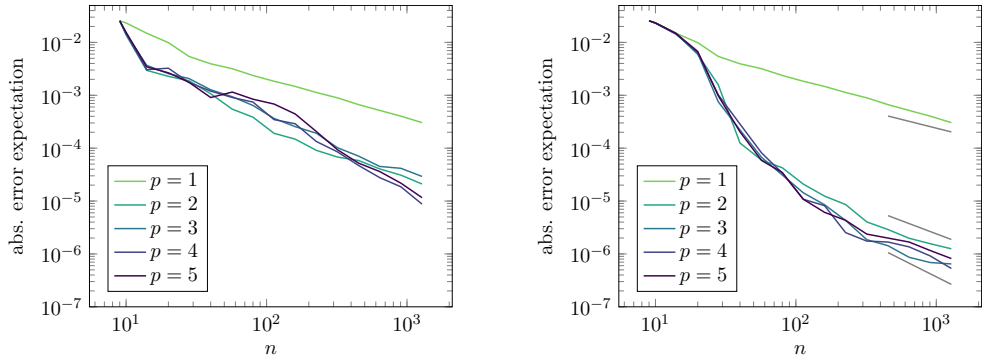
5.3.1 Function Approximation and Expected Value

In addition to the first two uncertain parameters, we now add a third one. The power x_3 of the withdrawn gas at the marked demand node is uniformly varied between 230 MW and 250 MW. See Figure 5.7(a) for the error of the original stochastic simplex collocation. The best convergence rate is obtained for a polynomial degree of $p = 2$ and increasing the degree results in a larger error estimate. This is not the case for our modified simplex stochastic collocation, see 5.7(b). The error is in the same order of magnitude for all polynomial degrees $p = 2, 3, 4, 5$ and converges with a rate of $\mathcal{O}(-1)$. Here, the good pre-asymptotic behavior can be seen even better than in $d = 2$ dimensions. Figure 5.8 shows the convergence results for the expectation. As in $d = 2$ dimensions, the reference value is computed with the new version of the simplex stochastic collocation, a polynomial degree of five, and $n = 5120$ interpolation points. For the original version (a), the difference between different polynomial degrees is not as large as predicted by the error estimator. The rate is of the same order of magnitude as for the new simplex stochastic collocation (b), but the new version benefits from the explicit kink approximation in the pre-asymptotic. Hence, only $m \approx 50$ instead of $m \approx 200$ sampling points are necessary to obtain an error of 10^{-4}.



(a) Original SSC (b) New SSC

Figure 5.7: $d = 3$. The l_1 error evaluated at 10^6 random points versus the number n of interpolation points with l_1 error estimator $\tilde{\varepsilon}_j$ for the original SSC without kink information (a), and for the new version with kink information (b).



(a) Original SSC (b) New SSC

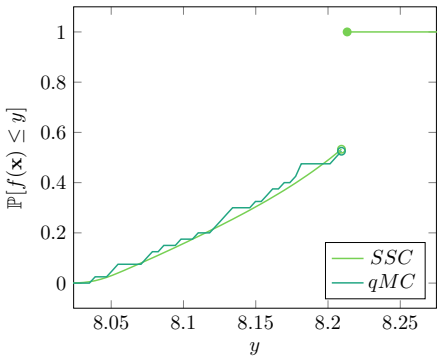
Figure 5.8: $d = 3$. The absolute error in the expected value versus the number n of interpolation points with l_1 error estimator $\tilde{\varepsilon}_j$ for the original SSC without kink information (a), and for the new version with kink information (b).

5.3.2 The Cumulative Density Function

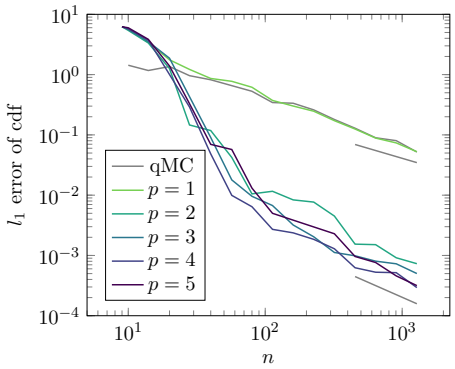
As in $d = 2$ dimensions, we calculate the cumulative density function. Again, we discretize the value range

$$I = \left[\min_{\mathbf{x}\in\Omega} f(\mathbf{x}),\ \max_{\mathbf{x}\in\Omega} f(\mathbf{x})\right]$$

with 50 equidistant nodes y_i and calculate $\mathbb{P}[f(\mathbf{x}) \leq y_i]$ by evaluating our approximation, and not the function itself, at 10^{12} Monte Carlo sampling points. Between the nodes y_i, the cumulative density function is approximated by a piecewise linear interpolation. See Figure 5.9(a) for a plot of the cumulative density function computed by using the simplex stochastic collocation with only $m = 40$ evaluation points. For the polynomial degrees $p = 2, \ldots, 5$ we get a result which is optically identical to the reference solution computed with the stochastic simplex collocation and $m = 5120$ sampling points. In contrast, the cumulative density function computed with $m = 40$ Halton points looks significantly worse. In Figure 5.9(b) the corresponding convergence rates are plotted. The error is nearly the same for the Halton sequence and for simplex stochastic collocation with linear polynomials. As in the case of the expected value, there is no benefit of using higher order polynomials but, again, the pre-asymptotic is very good and the obtained errors are significantly smaller than for the Halton sequence. Here, the attained convergence rate is approximately $\mathcal{O}(-1)$ as it was for the expectation.



(a) CDF



(b) Convergence plot

Figure 5.9: Shown is a plot of the cumulative density function in $d = 3$ dimensions for SSC and qMC (a), and the corresponding convergence rates (b). Both approximations of the cumulative density function are computed by evaluating the function at $m = 40$ points.

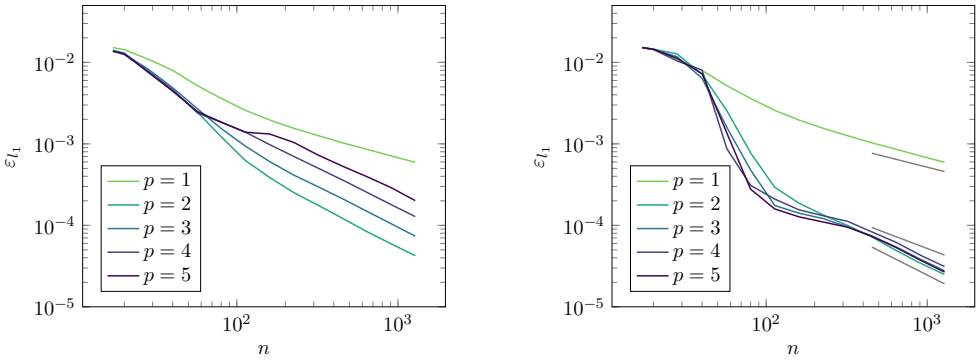
5.4 Input Uncertainties in Four Dimensions

5.4.1 Function Approximation and Expected Value

Lastly, we add an uncertainty at the power x_4 of the withdrawn gas at the third marked demand node. The power uniformly varies between 10 MW and 30 MW. As in $d = 2$ and $d = 3$ dimensions, the estimated l_1 error of the original simplex stochastic collocation increases with increasing polynomial degree, see Figure 5.10(a). This difference is no longer visible in the error of the expectation, where all polynomial degrees result in errors of same order of magnitude, see Figure 5.11(a). Again, the new version of the stochastic simplex collocation yields better results because of the better pre-asymptotic behavior. There is no visible benefit from using polynomials of degree $p \geq 3$, but the obtained convergence rates are of order $\mathcal{O}(-1)$. To achieve an error of 10^{-4}, we only need $m \approx 100$ sampling points, whereas the original version does not reach this error with $m \approx 1000$ sampling points.

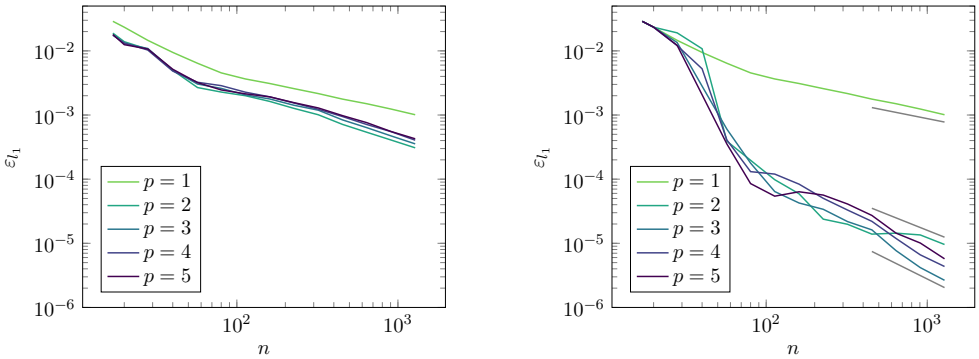
5.5 Comparison to Other Methods

Finally, we compare our new simplex stochastic collocation method with other common integration methods for computing an expected value. The convergence plots are shown in Figure 5.12 for dimensions $d = 2$, $d = 3$, and $d = 4$. The Monte Carlo quadrature does not make any requirements on the integrand, therefore, the theoretical convergence rate of $\mathcal{O}(-1/2)$ is obtained in all dimensions. The quasi-Monte Carlo quadrature rule with Halton points yields better results. In $d = 2$ and $d = 3$ dimensions a rate of approximately $\mathcal{O}(-1)$ is reached, whereas in $d = 4$ dimensions the rate is only $\mathcal{O}(-3/4)$. For sufficiently smooth integrands, sparse grid quadrature provides even better convergence. Since the considered integrand here is only in $\mathcal{C}^0(\Omega)$, it is quite interesting how well sparse grids perform. We use a regular and a spatially adaptive sparse grid with polynomials of degree five. In $d = 2$ dimen-



(a) Original SSC (b) New SSC

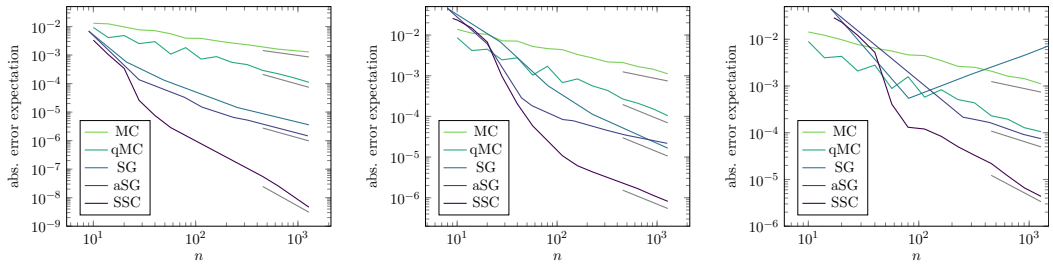
Figure 5.10: $d = 4$. The l_1 error evaluated at 10^6 random points versus the number n of interpolation points with l_1 error estimator $\tilde{\varepsilon}_j$ for the original SSC without kink information (a), and for the new version with kink information (b).



(a) original SSC (b) new SSC

Figure 5.11: $d = 4$. The absolute error in the expected value versus the number n of interpolation points with l_1 error estimator $\tilde{\varepsilon}_j$ for the original SSC without kink information (a), and for the new version with kink information (b).

sions, both sparse grids yield better results than the quasi-Monte Carlo quadrature, but with the same convergence rate of $\mathcal{O}(-1)$. Here, the spatially adaptive sparse grid is slightly better than the regular one. In $d = 3$ dimensions, both sparse grid quadratures are still better than the quasi-Monte Carlo quadrature but in the end, the adaptively added points yield worse results. In $d = 4$ dimensions, the regular sparse grid completely fails, and the spatially adaptive sparse grid is only as good as the quasi-Monte Carlo quadrature. In all dimensions, we get the best results with the simplex stochastic collocation. In $d = 2$ dimensions the maximal obtained convergence rate of $\mathcal{O}(-2)$ is twice as good as the one for sparse grids and quasi-Monte Carlo quadrature. Additionally, the pre-asymptotic is also better. In $d = 3$ and $d = 4$ dimensions, the convergence rate of the simplex stochastic collocation is the same, but, again, the better pre-asymptotic makes a difference. Concluding, we can say that the explicit kink approximation is useful and worthwhile, even though the theoretical convergence rates are not obtained due to the jumps in the second derivative. All methods requiring a certain smoothness suffer from these jumps.



(a) $d = 2$ (b) $d = 3$ (c) $d = 4$

Figure 5.12: The absolute error of the expected value versus the number n of interpolation points for Monte Carlo integration, quasi-Monte Carlo integration with the Halton sequence, regular sparse grids, spatially adaptive sparse grids, and simplex stochastic collocation with a polynomial degree of five.

6 Conclusion

This chapter provides a short summary of the results of this thesis. Furthermore, we point out open questions and discuss possible directions for future research.

Summary

In this thesis we considered methods for uncertainty quantification in gas network simulation. First, we derived a model for simulating the gas flow through a network consisting of several elements. We described the Euler equations for pipes, model equations for other types of elements, and mentioned their impact on the smoothness of the solution.

Next, we gave an overview over common methods for uncertainty quantification, including intrusive stochastic Galerkin methods and non-intrusive stochastic collocation methods. For each method we discussed assumptions, convergence rates, advantages and disadvantages, and possible applications. We saw that all these methods either assume sufficiently smooth functions or only provide poor convergence rates.

Because of this, we introduced the non-intrusive method of simplex stochastic collocation which allowed us to use an existing gas network solver. We hoped that the simplex stochastic collocation, originally intended for functions with jumps, would yield better convergence results than common methods for locally smooth functions with kinks. We described the idea of simplex stochastic collocation for a piecewise approximation on simplices of a function with polynomials of degree p. In the original version, to detect a discontinuity, the polynomial degree was reduced by one if the maximum or minimum of a polynomial approximation did not lie in the corners of the simplex. This assumption resulted in a finer triangulation near jumps or kinks, but we were not able to obtain the theoretical convergence rates of $\mathcal{O}(-(p+1)/d)$ with this version. By using the a-posteriori information whether a pressure control valve is active or not, we computed two approximations, one at each side of the kink, and used their maximum or minimum, respectively. By doing so, we could explicitly approximate the kink, which yields significantly better results. We proved that this modification results in algebraic convergence rates of $\mathcal{O}(-(p+1)/d)$ and verified the

rates with test functions in $d = 2, 3, 4$ dimensions. Moreover, we introduced two new error estimators for an adaptive refinement. We showed that in contrast to the original error estimators, our ones were reliable and solution-based without incorporating unnecessary simulation runs. Since multiple refinements had been proposed in [WI12a, WI12b, WI13], we analyzed the error distribution over the simplices and showed that multiple refinements are reasonable and do not affect the convergence rates.

The next essential step was to apply our new version of simplex stochastic collocation to a real gas network. As an example, we first presented two types of model errors to give an idea in what order of magnitude they are and which accuracy we had to reach with our method. We assumed an uncertain outgoing pressure at one control valve and up to three uncertain amounts of gas withdrawals at different demand nodes. For our quantity of interest, the outgoing pressure at another control valve, we computed the l_1 error estimator of the approximation, the absolute error of the expected value, and the l_1 error of the cumulative density function. In none of the cases, we could reach the desired convergence rates. We analyzed the problem in detail and discovered that the solution of the gas network solver was not as smooth as expected, since it has jumps in the second partial derivative due to numerical reasons. But, nevertheless, we saw that even in $d = 4$ dimensions maximally only 100 sampling points were necessary to approximate an expected value as accurate as the model error of 10^{-4}. A comparison with other common methods, such as sparse grid and (quasi-) Monte Carlo quadrature, showed that all methods suffer from jumps in the second partial derivatives and that our method benefits from the explicit kink approximation and, hence, yields significantly better results.

Outlook

So far, we have used the simplex stochastic collocation only for random variables that were uniformly distributed. Therefore, the next canonical step will be to extend the method of simplex stochastic collocation for random variables following other distributions with bounded support. Instead of weighting an error estimator with the area of a simplex, the error estimator could be weighted with the probability of a simplex. This idea was already presented for the original version of stochastic simplex collocation [WI12a, WI12b, WI13] and should not cause any problems. The more exciting question is whether simplex stochastic collocation can be used for random variables whose density function has unlimited support and how bounding the support influences the method.

Furthermore, we have seen that the method of Voronoi piecewise surrogate models provided better convergence results than simplex stochastic collocation for functions with many local minima and maxima. This could be due to the fact that, in Voronoi piecewise surrogate models, the approximation is based on solving a regression problem over the $2P$-nearest neighbors of each cell instead of solving an interpolation problem over the P-nearest neighbors. Therefore, it should be investigated how the use of regression affects simplex stochastic collocation and whether it improves it's convergence.

Acknowledgement

I would like to thank all the people who helped me to complete this thesis. First of all, I would like to thank my advisor Prof. Dr. Jochen Garcke for introducing me to this topic. The excellent working conditions and the opportunity to participate in the scientific community at several scientific conferences were an essential step towards this thesis. Furthermore, I would like to thank the committee member Prof. Dr. Marc Alexander Schweitzer for providing the second opinion on this dissertation. Moreover, I would like to thank Katrin and, especially, Erdem who carefully read the whole thesis. I am also thankful to the Fraunhofer staff, who helped me to resolve problems with the MYNTS software. Of course, all numerical simulations would have been impossible without the great students and Ralph Thesen from our IT-Team who maintained the workstations and the cluster system at the Institute for Numerical Simulation. Thanks a lot.
Last but not least, I thank my family for all their support and encouragement. I would like to express my deepest gratitude to Erdem for all his never-ending encouragement and support in every situation.

Bibliography

[AFM06] P. Ailliot, E. Frénod, and V. Monbet. Long term object drift forecast in the ocean with tide and wind. *Multiscale Modeling & Simulation*, 5(2):514–531, 2006.

[AGP+08] F. Augustin, A. Gilg, M. Paffrath, P. Rentrop, and U. Wever. A survey in mathematics for industry: Polynomial chaos for the approximation of uncertainties: chances and limits. *European Journal of Applied Mathematics*, 19:149–190, 2008.

[Ale01] C. Alexander. *Market models: a guide to financial data analysis*. John Wiley & Sons, Chichester, 2001.

[AV13] P. Attar and P. Vedula. On convergence of moments in uncertainty quantification based on direct quadrature. *Reliability Engineering & System Safety*, 111:119–125, 2013.

[BD17] M. Beres and S. Domesova. The stochastic Galerkin method for Darcy flow problem with log-normal random field coefficients. *Advances in Electrical and Electronic Engineering*, 15(2):267–279, 2017.

[Bea07] P. Beater. *Pneumatic Drives: System Design, Modelling and Control*. Springer, Berlin, 2007.

[BNT07] I. Babuška, F. Nobile, and R. Tempone. A stochastic collocation method for elliptic partial differential equations with random input data. *SIAM Journal on Numerical Analysis*, 45(3):1005–1034, 2007.

[Boy82] J. Boyd. A Chebyshev polynomial rate-of-convergence theorem for Stieltjes function. *Mathematics of Computation*, 39(159):201–206, 1982.

[Bro02] G. Brown. The history of the Darcy-Weisbach equation for pipe flow resistance. In *Environmental and Water Resources History*, pages 34–43. American Society of Civil Engineers, 2002.

[BTNT12] J. Beck, R. Tempone, F. Nobile, and L. Tamellini. On the optimal polynomial approximation of stochastic PDEs by Galerkin and collocation methods. *Mathematical Models and Methods in Applied Sciences*, 22(09):1250023, 2012.

[CC60] C. Clenshaw and A. Curtis. A method for numerical integration on an automatic computer. *Numerische Mathematik*, 2(1):197–205, 1960.

[CCH+16] T. Clees, K. Cassirer, N. Hornung, B. Klaassen, I. Nikitin, L. Nikitina, R. Suter, and I. Torgovitskaia. MYNTS: Multi-physics network simulator. In *Proceedings of the 6th International Conference on Simulation and Modeling Methodologies, Technologies and Applications*, pages 179–186, 2016.

[CCMHD16] A. Cagatay Cobanoglu, S. Mößner, M. Hojjat, and F. Duddeck. Model order reduction methods for explicit FEM. In *Conference: Science in the Age of Experience*, pages 179–186, 2016.

[CE09] F. Calabrò and A. Corbo Esposito. An evaluation of Clenshaw–Curtis quadrature rule for integration w.r.t. singular measures. *Journal of Computational and Applied Mathematics*, 229(1):120–128, 2009.

[CGP17] D. Crevillén-García and H. Power. Multilevel and quasi-Monte Carlo methods for uncertainty quantification in particle travel times through random heterogeneous porous media. *Royal Society Open Science*, 4(8):170–203, 2017.

[CGST11] K. Cliffe, M. Giles, R. Scheichl, and A. Teckentrup. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, 14(1):3–15, 2011.

[Che67] P. Chebyshev. On mean values (english translation). *Mathmaticheskii Sbornik*, 2(1):1–9, 1867.

[CM47] R. Cameron and W.T. Martin. The orthogonal development of nonlinear functionals in series of Fourier-Hermite functionals. *Annals of Mathematics*, 48(2):385–392, 1947.

[CMM09] Th. Crestaux, O. Le Maitre, and J.-M. Martinez. Polynomial chaos expansion for sensitivity analysis. *Reliability Engineering & System Safety*, 94(7):1161–1172, 2009. Special Issue on Sensitivity Analysis.

[CQ15] P. Chen and A. Quarteroni. A new algorithm for high-dimensional uncertainty quantification based on dimension-adaptive sparse grid approximation and reduced basis methods. *Journal of Computational Physics*, 298(C):176–193, 2015.

[Cra09] Crane Co. Engineering Department. *Crane Technical Paper 410M: Flow of Fluids Through Valves, Fittings and Pipe*. 2009.

[DB12] F. Dullien and H. Brenner. *Porous Media: Fluid Transport and Pore Structure*. Elsevier Science, 2012.

[Des87] A. Desbarats. Numerical estimation of effective permeability in sand-shale formations. *Water Resources Research*, 23(2):273–286, 1987.

[Dev86] L. Devroye. *Non-uniform random variate generation*. Springer, New York, 1986.

[DG13] M. D'Elia and M. Gunzburger. Coarse-grid sampling interpolatory methods for approximating Gaussian random fields. *SIAM/ASA Journal on Uncertainty Quantification*, 1(1):270–296, 2013.

[Duc05] W. Duckett. Risk analysis and the acceptable probability of failure. *Structural Engineer*, 83:25–26, 2005.

[Dud08] F. Duddeck. Multidisciplinary optimization of car bodies. *Structural and Multidisciplinary Optimization*, 35(4):375–389, 2008.

[DW76] H. Daneshyar and W. Woods. *One-Dimensional Compressible Flow*. Pergamon Press, 1st edition, 1976.

[DW81] C. Dunham and J. Williams. Rate of convergence of discretization in Chebyshev approximation. *Mathematics of Computation*, 37(155):135–139, 1981.

[EB09] M. Eldred and J. Burkardt. Comparison of non-intrusive polynomial chaos and stochastic collocation methods for uncertainty quantification. In *47th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2009.

[ES14] O. Ernst and B. Sprungk. Stochastic collocation for elliptic PDEs with random data: The lognormal case. In J. Garcke and D. Pflüger, editors, *Sparse Grids and Applications - Munich 2012*, pages 29–53. Springer International Publishing, 2014.

[Ete81] N. Etemadi. An elementary proof of the strong law of large numbers. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 55(1):119–122, 1981.

[FP16] F. Franzelin and D. Pflüger. From data to uncertainty: An efficient integrated data-driven sparse grid approach to propagate uncertainty. In J. Garcke and D. Pflüger, editors, *Sparse Grids and Applications - Stuttgart 2014*, pages 29–49. Springer International Publishing, 2016.

[Gar07] J. Garcke. A dimension adaptive sparse grid combination technique for machine learning. In W. Read, J. Larson, and A. Roberts, editors, *Proceedings of the 13th Biennial Computational Techniques and Applications Conference, CTAC-2006*, volume 48 of *ANZIAM J.*, pages C725–C740, 2007.

[Gar12] J. Garcke. A dimension adaptive combination technique using localised adaptation criteria. In H. Bock, X. Hoang, R. Rannacher, and J. Schlöder, editors, *Modeling, Simulation and Optimization of Complex Processes*, pages 115–125. Springer Berlin Heidelberg, 2012.

[Gar13] J. Garcke. Sparse grids in a nutshell. In J. Garcke and M. Griebel, editors, *Sparse grids and applications*, volume 88 of *Lecture Notes in Computational Science and Engineering*, pages 57–80. Springer, 2013.

[GF16] D. Grunert and J. Fehr. Identification of nonlinear behavior with clustering techniques in car crash simulations for better model reduction. *Advanced Modeling and Simulation in Engineering Sciences*, 3(1):20, 2016.

[GG98] T. Gerstner and M. Griebel. Numerical integration using sparse grids. *Numerical Algorithms*, 18:209–232, 1998.

[GG03] T. Gerstner and M. Griebel. Dimension–adaptive tensor–product quadrature. *Computing*, 71(1):65–87, 2003.

[GI17] S. Ghili and G. Iaccarino. Least squares approximation of polynomial chaos expansions with optimized grid points. *SIAM Journal on Scientific Computing*, 39(5):A1991–A2019, 2017.

[Gil15] M. Giles. Multilevel Monte Carlo methods. *Acta Numerica*, 24:259–328, 2015.

[GK14] J. Garcke and I. Klompmaker. Adaptive sparse grids in reinforcement learning. In S. Dahlke, W. Dahmen, M. Griebel, W. Hackbusch, K. Ritter, R. Schneider, C. Schwab, and H. Yserentant, editors, *Extraction of Quantifiable Information from Complex Systems*, volume 102 of *Lecture Notes in Computational Science and Engineering*, pages 179–194. Springer, 2014.

[GKS13] M. Griebel, F. Kuo, and I. Sloan. The smoothing effect of integration in $\mathbb{R}^d$ and the ANOVA decomposition. *Mathematics of Computation*, 82:383–400, 2013.

[GKS17] M. Griebel, F. Kuo, and I. Sloan. Note on "The smoothing effect of integration in $\mathbb{R}^d$ and the ANOVA decomposition". *Mathematics of Computation*, 86:1855–1876, 2017.

[GKW+07] B. Ganis, H. Klie, M. Wheeler, T. Wildey, I. Yotov, and D. Zhang. Stochastic collocation and mixed finite elements for flow in porous media. Preprint available at https://www.mathematics.pitt.edu/sites/default/files/research-pdfs/smfe.pdf, 2007.

[Gla13] P. Glasserman. *Monte Carlo Methods in Financial Engineering*. Stochastic Modelling and Applied Probability. Springer New York, 2013.

[Hal60] J. Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematic*, 2:84–90, 1960.

[Hof73] P. Hofer. Beurteilung von Fehlern in Rohrnetzberechnungen. *GWF Gas/Erdgas*, 11:113–119, 1973.

[INC01] Inland Navigation Commission Working Group 19. Ship collisions due to the presence of bridges. Technical report, PIANC – The World Association for Waterborne Transport Infrastructure, 2001.

[JAX11] J. Jakeman, R. Archibald, and D. Xiu. Characterization of discontinuities in high-dimensional stochastic problems on adaptive sparse grids. *Journal of Computational Physics*, 230(10):3977–3997, 2011.

[JC12] G. Jones and S. Chapman. Modeling growth in biological materials. *SIAM Review*, 54(1):52–118, 2012.

[JEX10] J. Jakeman, M. Eldred, and D. Xiu. Numerical approach for quantification of epistemic uncertainty. *Journal of Computational Physics*, 229(12):4648–4663, 2010.

[KBJ14] N. Kantas, A. Beskos, and A. Jasra. Sequential Monte Carlo methods for high-dimensional inverse problems: A case study for the Navier–Stokes equations. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):464–489, 2014.

[KBK+08] R. Kerr, T. Bartol, B. Kaminsky, M. Dittrich, J.-C. Chang, S. Baden, T. Sejnowski, and J. Stiles. Fast Monte Carlo simulation methods for biological reaction-diffusion systems in solution and on surfaces. *SIAM Journal on Scientific Computing*, 30(6):3126–3149, 2008.

[Ker71] D. Kershaw. A note on the convergence of interpolatory cubic splines. *SIAM Journal on Numerical Analysis*, 8:67–74, 03 1971.

[KF09] D. Koller and N. Friedman. *Probabilistic Graphical Models*. The MIT Press, 2009.

[Khi29] A. Khinchin. Sur la loi des grandes nombres. *Comptes Rendus de l'Academie des Sciences*, 188, 1929.

[KHPS15] T. Koch, B. Hiller, M. Pfetsch, and L. Schewe. *Evaluating Gas Network Capacities*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2015.

[KK05] W. Knorr and J. Kattge. Inversion of terrestrial ecosystem model parameter values against eddy covariance measurements by Monte Carlo sampling. *Global Change Biology*, 11(8):1333–1351, 2005.

[KKK10] R. Korn, E. Korn, and G. Kroisandt. *Monte Carlo Methods and Models in Finance and Insurance*. Chapman & Hall/CRC Financial Mathematics Series. CRC Press, 2010.

[KKSV00] J. Kaipio, V. Kolehmainen, E. Somersalo, and M. Vauhkonen. Statistical inversion and Monte Carlo sampling methods in electrical impedance tomography. *Inverse Problems*, 16(5):1487–1522, 2000.

[Kol30] A. Kolmogorov. Sur la loi forte des grandes nombres. *Comptes Rendus de l'Academie des Sciences*, 191:910–911, 1930.

[Kol33] A. Kolmogorov. *Grundbegriffe der Wahrscheinlichkeitsrechnung.* Springer, Berlin, 1933.

[KW16] J. Ko and H. Wynn. The algebraic method in quadrature for uncertainty quantification. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):331–357, 2016.

[Lar93] O. Larsen. *Ship collision with bridges: the interaction between vessel traffic and bridge structures.* Number 4 in Structural engineering documents. International Association for Bridge and Structural Engineering, 1993.

[LGBD+18] Y. Le Guennec, J.-P. Brunet, F. Daim, M. Chau, and Y. Tourbier. A parametric and non-intrusive reduced order model of car crash simulation. *Computer Methods in Applied Mechanics and Engineering*, 2018.

[LMQR13] T. Lassila, A. Manzoni, A. Quarteroni, and G. Rozza. A reduced computational and geometrical framework for inverse problems in haemodynamics. *International Journal for Numerical Methods in Biomedical Engineering*, 29(7):741–776, 2013.

[Lur08] M. Lurie. *Modeling of Oil Product and Gas Pipeline Transportation.* Wiley-VCH Verlag GmbH and Co. KGaA, Weinheim, Germany, 2008.

[LZ07] H. Li and D. Zhang. Probabilistic collocation method for flow in porous media: Comparisons with other stochastic methods. *Water Resources Research*, 43(9), 2007.

[Men97] J. Menzies. *Bridge failures, hazards and societal risk*, pages 36–41. Thomas Telford Ltd, 1997.

[Mer09] J. Mercer. XVI. Functions of positive and negative type, and their connection the theory of integral equations. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 209(441-458):415–446, 1909.

[MK10] O. Le Maitre and O. Knio. *Spectral Methods for Uncertainty Quantification.* Springer Netherlands, 2010.

[Nie92] H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods.* Society for Industrial and Applied Mathematics, 1992.

[Nik33] J. Nikuradse. Strömungsgesetze in rauhen Rohren. *VDI-Forschungsheft*, 1933.

[Nov11] M. Novelinková. Comparison of Clenshaw-Curtis and Gauss quadrature. In *WDS 2011 - Proceedings of Contributed Papers, Part I*, pages 67–71, 2011.

[Nyq28] H. Nyquist. Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 47:617–644, 1928.

[Osi84] A. Osiadacz. Simulation of transient gas flows in networks. *International Journal for Numerical Methods in Fluids*, 4:13–24, 1984.

[Owe06] A. Owen. Halton sequences avoid the origin. *SIAM Review*, 43(3):487–503, 2006.

[Pag12] H. Paganetti. Range uncertainties in proton therapy and the role of Monte Carlo simulations. *Physics in Medicine & Biology*, 57(11):R99–R117, 2012.

[Pap68] J. Papay. A Termelestechnologiai Parameterek Valtozasa a Gazlelepk Muvelese Soran. *OGIL MUSZ, Tud, Kuzl., Budapest*, pages 267–273, 1968.

[Pfl10] D. Pflüger. *Spatially Adaptive Sparse Grids for High-Dimensional Problems*. Verlag Dr. Hut, München, 2010.

[Pfl12] D. Pflüger. Spatially adaptive refinement. In J. Garcke and M. Griebel, editors, *Sparse Grids and Applications*, Lecture Notes in Computational Science and Engineering, pages 243–262, Berlin Heidelberg, 2012. Springer.

[PNI15] M. Pettersson, J. Nordström, and G. Iaccarino. *Polynomial Chaos Methods for Hyperbolic Partial Differential Equations: Numerical Techniques for Fluid Dynamics Problems in the Presence of Uncertainties*. Springer International Publishing, 2015.

[PVV11] M. Perego, A. Veneziani, and C. Vergara. Variational approach for estimating the compliance of the cardiovascular tissue: an inverse fluid-structure interaction problem. *SIAM Journal on Scientific Computing*, 33(3):1181–1211, 2011.

[QBS+06] C. Quick, D. Berger, R. Stewart, G. Laine, C. Hartley, and A. Noordergraaf. Resolving the hemodynamic inverse problem. *IEEE Transactions on Biomedical Engineering*, 53(3):361–368, 2006.

[RMR88] Y. Rudy and B. Messinger-Rapport. The inverse problem in electrocardiography: solutions in terms of epicardial potentials. *Critical reviews in biomedical engineering*, 16(3):215–268, 1988.

[RND+12a] F. Rizzi, H. Najm, B. Debusschere, K. Sargsyan, M. Salloum, H. Adalsteinsson, and O. Knio. Uncertainty quantification in MD simulations. Part I: Forward propagation. *Multiscale Modeling & Simulation*, 10(4):1428–1459, 2012.

[RND+12b] F. Rizzi, H. Najm, B. Debusschere, K. Sargsyan, M. Salloum, H. Adalsteinsson, and O. Knio. Uncertainty quantification in MD simulations.

Part II: Bayesian inference of force-field parameters. *Multiscale Modeling & Simulation*, 10(4):1428–1459, 2012.

[RSP+17] A. Rushdi, L. Swiler, E. Phipps, M. D'Elia, and M. Ebeida. VPS: Voronoi piecewise surrogate models for high-dimensional data fitting. *International Journal for Uncertainty Quantification*, 7(1):1–21, 2017.

[SBS00] W. Schneider, T. Bortfeld, and W. Schlegel. Correlation between CT numbers and tissue parameters needed for Monte Carlo simulations of clinical dose distributions. *Physics in Medicine & Biology*, 45(2):459–478, 2000.

[Sha49] C. Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21, 1949.

[Smo63] S. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. *Soviet Mathematics, Doklady*, 4:240–243, 1963.

[Sob58] I. Sobol. Pseudo-random numbers for the machine Strela (english translation). *Theory of Probability & Its Applications*, 3:192–197, 1958.

[SST17] R. Scheichl, A. Stuart, and A. Teckentrup. Quasi-Monte Carlo and multilevel Monte Carlo methods for computing posterior expectations in elliptic inverse problems. *SIAM/ASA Journal on Uncertainty Quantification*, 5:493–518, 2017.

[SSW16] M. Schmidt, M. Steinbach, and B. Willert. High detail stationary optimization models for gas networks: validation and results. *Optimization and Engineering*, 17(2):437–472, 2016.

[Sto06] J. Stoer. *Numerische Mathematik 1.* Springer-Lehrbuch. Springer Berlin Heidelberg, 2006.

[Sud08] B. Sudret. Global sensitivity analysis using polynomial chaos expansions. *Reliability Engineering & System Safety*, 93(7):964 – 979, 2008. Bayesian Networks in Dependability.

[Sul15] T. Sullivan. *Introduction to Uncertainty Quantification.* Texts in Applied Mathematics. Springer International Publishing, 2015.

[SX95] T. Sauer and Y. Xu. A case study in multivariate Lagrange interpolation. In S. Singh, editor, *Approximation Theory, Wavelets and Applications*, pages 443–452. Springer, Dordrecht, 1995.

[TGG13] T. Thorarinsdottir, T. Gneiting, and N. Gissibl. Using proper divergence functions to evaluate climate models. *SIAM/ASA Journal on Uncertainty Quantification*, 1(1):522–534, 2013.

[TI14] G. Tang and G. Iaccarino. Subsampled Gauss quadrature nodes for estimating polynomial chaos expansions. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):423–443, 2014.

[TMEP11] R. Tuminaro, Ch. Miller, H. Elman, and E. Phipps. Assessment of collocation and Galerkin approaches to linear diffusion equations with random data. *International Journal for Uncertainty Quantification*, 1(1):19–33, 2011.

[Tre08] L. Trefethen. Is Gauss quadrature better than Clenshaw–Curtis? *SIAM Review*, 50(1):67–87, 2008.

[WI12a] J. Witteveen and G. Iaccarino. Refinement criteria for simplex stochastic collocation with local extremum diminishing robustness. *SIAM Journal on Scientific Computing*, 34(3), 2012.

[WI12b] J. Witteveen and G. Iaccarino. Simplex stochastic collocation with random sampling and extrapolation for nonhypercube probability spaces. *SIAM Journal on Scientific Computing*, 34(2):814–838, 2012.

[WI13] J. Witteveen and G. Iaccarino. Simplex stochastic collocation with ENO-type stencil selection for robust uncertainty quantification. *Journal of Computational Physics*, 239:1–21, 2013.

[Wie38] N. Wiener. The homogeneous chaos. *American Journal of Mathematics*, 60(4):897–936, 1938.

[Xiu10] D. Xiu. *Numerical Methods for Stochastic Computations: A Spectral Method Approach*. Princeton University Press, Princeton, NJ, USA, 2010.

[ZL04] D. Zhang and Z. Lu. An efficient, high-order perturbation approach for flow in random porous media via Karhunen-Loève and polynomial expansions. *Journal of Computational Physics*, 194(2):773–794, 2004.