Institut für Numerische Simulation
Rheinische Friedrich-Wilhelms-Universität Bonn

Wegelerstraße 6 • 53115 Bonn • Germany
phone +49 228 73-3427 • fax +49 228 73-7527
www.ins.uni-bonn.de

M. Griebel, J. Hamaekers

Molecular Dynamics Simulations of the Mechanical Properties of Polyethylene-Carbon Nanotube Composites

INS Preprint No. 0502

May 2005

Table of Contents

1 Introduction **2**

2 Computational Methods **4**
2.1 Molecular Dynamics Simulations 4
2.1.1 (N, P, T) Ensemble 6
2.1.2 Periodic Boundary Conditions 12
2.1.3 Time Integration 17
2.1.4 Local Minimization 20
2.2 Potentials for Hydrocarbons 23
2.2.1 Reactive Atomistic Model (Model I) 23
2.2.2 United-Atom Model (Model II) 30
2.2.3 Efficient Implementation 32
2.2.4 Parallel Implementation 39
2.3 Elastic Moduli and Constants 42

3 Numerical Experiments **44**
3.1 Equilibrated Systems 45
3.1.1 Carbon Nanotubes 45
3.1.2 Polyethylene Matrices 47
3.1.3 Composites 47
3.1.4 Functionalized Nanotube Composites 50
3.2 Simulation of Tensile Load Tests 50

4 Discussion **53**

5 Concluding Remarks **56**

1 Introduction

With the discovery of pure carbon structures different from graphite and diamond, a new and fast-developing area in nanomaterial science began [1]. In particular, tube-like structures were first reported by Iijima in 1991; see [2]. The diameter of these carbon nanotubes is in the nanometer range, but their length can be in the micrometer range [3]. Meanwhile, their mechanical properties have been studied by theoretical predictions and experimental measurements [4, 5]. Due to their structure and form, they tolerate extreme distortion without fracture. They also show elastic bending, twisting, buckling, and other reversible deformations. The bending stiffness is in the range of 0.4 TPa to 4 TPa [6]; furthermore a local tension of hundreds of gigapascals can be reached before fracture occurs [7]. Due to these properties, carbon nanotubes can be used to reinforce polymer composites. Here, they possess the potential for large increases in strength and stiffness in comparison to conventional carbon-fiber-reinforced polymer composites. Thus, nanotube-polymer composites have gained considerable interest in the materials research community.

Meanwhile, some nanotube composite materials have been characterized experimentally [8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18]. This is however a demanding and expensive task. To this end, computational methods can be used to greatly facilitate the development of nanotube composite materials. Computer simulations allow for parametric studies of the influence of composite and geometry on the material properties. In particular, *first-principle* techniques [19, 20, 21, 22, 23], *semi-empirical* schemes [24], and *empirical potential* methods [19, 21, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34] have been applied successfully to study nanostructures. Here, especially for large systems with thousands of atoms and more, molecular dynamics simulations that use empirical potentials are an important tool to better understand the properties of polymer-carbon nanotube composites. In this work, we derive stress-strain curves from molecular dynamics simulations of polymer-carbon nanotube composites to predict their macroscopic elastic moduli. We then compare these results to that obtained with a *rule-of-mixtures*, which takes only the volume fraction of the fibers into account, and an *extended rule-of-mixtures*, which takes also the distribution of the fibers into account [35, 36]. Furthermore, we study the effect of cross-links between the matrix and the nanotube on the mechanical properties of the composite material.

In several earlier works, molecular dynamics simulations have been successfully applied to predict elastic properties of polymer-carbon nanotube composites [37, 38, 39]. To this end, the application of strain to a sample is a major requisite. In the most simple approach, this can be accomplished by uniformly expanding the length of the simulation cell in the direction of the deformation. To equilibrate the system for the new cell size, the coordinates of the atoms are rescaled to fit to the new geometry, and a molecular dynamics simulation or a potential energy minimization is performed. This way, the stress at different strain values can be calculated successively. For further details, see [40, 19, 25, 27, 24, 31, 32, 41]. Besides, there exist two methods to apply strain, which are based on the Parrinello-Rahman [42] approach, namely the so-called fluctuation method to calculate elastic constants [43, 44, 45, 46, 47, 48, 49, 41] and the dynamic method. The dynamic method uses constant stress molecular dynamics to measure the stress-strain behavior of a material subjected to an applied load [50, 51, 46, 52, 53].

In the current work, we carry out the application of strain by employing a Parrinello-Rahman-Nosé Lagrangian to control stress and temperature in an (N, P, T) ensemble [42, 54, 55, 56, 57, 58]. Similar to the dynamic method, we apply an external stress tensor within the equations of motion and measure the linear stress tensor. We compute the stress-strain curves of different model systems, a periodically replicated uncapped $(10, 10)$ single-walled carbon nanotube, a finite $(10, 10)$ single-walled capped carbon nanotube, a functionalized finite $(10, 10)$ single-walled capped carbon nanotube embedded in a polyethylene matrix, and the polyethylene matrix itself. In the case of the fully embedded capped carbon nanotube, we use polyethylene matrices of different size, and in the case of the functionalized nanotube we consider systems with a different amount of cross-links. To model the bonded interaction within these hydrocarbon systems, we use a many-body bond order potential due to Brenner [59, 60]. In the case of the composite materials without cross-links, we model the polyethylene matrix by a united-atom potential [61] and use Brenner's potential just for the carbon nanotube. In both models, the non-bonded interaction of the atoms is represented by a simple Lennard-Jones potential. We exploit the slopes of the stress-strain curves to derive different elastic moduli and constants.

The remainder of this paper is organized as follows: In section 2 we discuss the computational methods that we use in our study of polymer-carbon nanotube composites. In section 2.1, we give the details of the molecular dynamics approach in the framework of an (N, P, T) ensemble. In section 2.2, we discuss two different models to represent hydrocarbon systems. In section 2.3, we describe a computational method to derive the elasticity tensor. Section 3 gives the results of our numerical experiments. We relate these results to two different rules-of-mixtures in section 4. Finally we give some concluding remarks.

2 Computational Methods

In this section, we describe the modeling and implementational aspects of our molecular dynamics approach to determine the elastic properties of polyethylene-carbon nanotube composites.

First, we give the basics of molecular dynamics in view of the simulation of nanotube composites. In particular, we discuss an isobaric-isothermal ensemble due to Parrinello-Rahman-Nosé and present some details on the application of periodic boundary conditions. Then, we discuss the discretization in time of the corresponding equations of motion. Furthermore, a conjugate gradient method to perform local energy minimization is described. It is needed to initialize the systems for a simulation run. Afterwards, we review the potentials we use to model hydrocarbon systems. Here, we give some details on our implementation to efficiently evaluate the potential energy and the forces on the particles. To deal with large systems over long times, a parallelization of the method is necessary. Here, we discuss a classical domain decomposition approach. Finally, we present an approach to generate stress-strain curves. From these, we then can derive elastic moduli and constants using the generalized Hooke's law.

2.1 Molecular Dynamics Simulations

We consider a system of N particles with masses $\{m_1, \cdots, m_N\}$, which is characterized by coordinates $\{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$ and momenta $\{\mathbf{p}_1, \cdots, \mathbf{p}_N\}$. Here, the coordinates $\mathbf{x} := (\mathbf{x}_1, \cdots, \mathbf{x}_N)^T$ and momenta $\mathbf{p} := (\mathbf{p}_1, \cdots, \mathbf{p}_N)^T$ are time-dependent vectors in $\mathbb{R}^{3N}$. The dynamics of the system is described by Hamilton's equations of motion

$$\dot{\mathbf{x}} = \nabla_{\mathbf{p}} \mathcal{H}(\mathbf{x}, \mathbf{p}) \,, \qquad \dot{\mathbf{p}} = -\nabla_{\mathbf{x}} \mathcal{H}(\mathbf{x}, \mathbf{p}) \,, \tag{1}$$

with a so-called Hamiltonian $\mathcal{H}$. We furthermore assume that the interaction of the particles is given by a conservative potential

$$U = U(\mathbf{x}) \quad \text{with} \quad \frac{\partial U}{\partial t} \equiv 0 \,.$$

Then, the Hamiltonian can be written in the form

$$\mathcal{H}(\mathbf{x}, \mathbf{p}) = \frac{1}{2} \sum_{i=1}^{N} \frac{\mathbf{p}_i^T \mathbf{p}_i}{m_i} + U(\mathbf{x}_1, \ldots, \mathbf{x}_N) \,, \tag{2}$$

with the kinetic energy $E_{\text{kin}} := \frac{1}{2} \sum_{i=1}^{N} \frac{\mathbf{p}_i^T \mathbf{p}_i}{m_i}$ and the potential energy $E_{\text{pot}} := U$. Moreover, Hamilton's equations of motion (1) are then equivalent to Newton's equations of motion together with the impulse relation $\mathbf{p}_i(t) = m_i \dot{\mathbf{x}}(t)$. Newton's equations of motion are given by Newton's second law

$$m_i \ddot{\mathbf{x}}_i = \mathbf{F}_i(\mathbf{x}, t) \,,$$

where the force acting on particle i is defined as

$$\mathbf{F}_i := -\nabla_{\mathbf{x}_i} U \,. \tag{3}$$

A Hamiltonian system that consists of the equations of motion (1) and the Hamiltonian (2) gives the general basis of molecular dynamics.

If we use Hamilton's equations of motion (1) to calculate the total time derivative of the Hamiltonian

$$\begin{aligned}\frac{d}{dt}\mathcal{H}(\mathbf{x},\mathbf{p}) &= \nabla_{\mathbf{x}}\mathcal{H}(\mathbf{x},\mathbf{p})\,\dot{\mathbf{x}} + \nabla_{\mathbf{p}}\mathcal{H}(\mathbf{x},\mathbf{p})\,\dot{\mathbf{p}} \\ &= \nabla_{\mathbf{x}}\mathcal{H}(\mathbf{x},\mathbf{p})\,\nabla_{\mathbf{p}}\mathcal{H}(\mathbf{x},\mathbf{p}) - \nabla_{\mathbf{p}}\mathcal{H}(\mathbf{x},\mathbf{p})\,\nabla_{\mathbf{x}}\mathcal{H}(\mathbf{x},\mathbf{p}) \equiv 0\,,\end{aligned}$$

we see that the total energy $E_{\text{tot}} = E_{\text{kin}} + E_{\text{pot}}$ is conserved over time. Therefore, we obtain the so-called microcanonical (N, V, E) ensemble, if we assume a simulation domain with a finite volume V and appropriate boundary conditions. Here, the number of particles, the volume, and the total energy are constant over time. Temperature and pressure are not constant over time. They, however, are thermodynamic quantities anyway, which are meaningful only at equilibrium [62]. If a microscopic dynamic variable α takes on instantaneous values $\alpha(t)$ along a trajectory, then the time average

$$\langle\alpha\rangle_t := \lim_{t\to\infty}\frac{1}{t}\int_0^t \alpha(\tau)\,d\tau$$

results in the related thermodynamic value due to the so-called ergodic hypothesis [63]. This dynamic variable can be any function of the coordinates and momenta of the particles of the system. The thermodynamic temperature T is related to the average of the kinetic energy E_{kin} by the equipartition theorem

$$\left\langle \frac{1}{2}\sum_{i=1}^{N}\frac{\mathbf{p}_i^T\mathbf{p}_i}{m_i}\right\rangle_t = \langle E_{\text{kin}}\rangle_t = \frac{N_f k_B T}{2}\,, \tag{4}$$

where N_f denotes the number of degrees of freedom of the physical system, and k_B is Boltzmann's constant. In an unrestricted system with N atoms, N_f is $3N$ because each atom has three velocity components. In particular, the thermodynamic temperature T is given by the average over time of the instantaneous temperature $T_{\text{inst}} := \frac{2}{N_f k_B}E_{\text{kin}}$. Furthermore, the thermodynamic pressure P is related to the thermodynamic temperature T, the volume V, and the internal virial $W = \langle W_{\text{inst}}\rangle_t$ by the equation of state

$$PV = \frac{1}{3}N_f k_B T + \langle W_{\text{inst}}\rangle_t\,, \tag{5}$$

which is derived from the classical virial theorem [62]. Here, the instantaneous internal virial W_{inst} is related to the fluctuations of the potential energy U with respect to the volume V by $\frac{W_{\text{inst}}}{V} = -\frac{dU}{dV}$. If we define the instantaneous pressure in the form

$$P_{\text{inst}} := \frac{2}{3V}E_{\text{kin}} + \frac{1}{V}W_{\text{inst}}\,, \tag{6}$$

then the thermodynamic pressure is also given by the average $P = \langle P_{\text{inst}}\rangle_t$. This way, we can perform molecular dynamics simulations for an (N, V, E) ensemble to obtain equilibrium thermodynamic properties like temperature and pressure. In particular, temperature and pressure are first-order equilibrium thermodynamic properties that are in the thermodynamical limit independent of the statistical ensemble used.

An ensemble is specified by the external conditions describing an equilibrium state. Beside the (N ,V ,E) or microcanonical ensemble, there is the (N, V, T) or canonical ensemble, the (N, P, T) or isothermal-isobaric ensemble, and the (μ, V, T) or grand canonical ensemble, where μ denotes the chemical potential. Here, the noted properties are constant.

Now, second-order equilibrium thermodynamic properties like elastic constants can be calculated using microscopic expressions in the form of fluctuations of first-order properties. But in contrast to these, second-order properties are dependent on the respective ensemble. Therefore, it is important to use the appropriate ensemble in a simulation. The transformation and relation between different ensembles has been discussed in the literature, see for example [64, 62, 65]. In the following, we present in detail the use of an extended Hamiltonian system. This corresponds to an isothermal-isobaric (N, P, T) ensemble, which is more convenient than the (N, V, E) or the (N, V, T) ensemble to determine mechanical properties of nanocomposites by molecular dynamics simulations.

2.1.1 (N, P, T) Ensemble

As known from classical mechanics textbooks [66, 67, 68, 69] the dynamics of a system can also be described by a Lagrangian

$$\mathcal{L}(\mathbf{x}, \mathbf{v}) = \frac{1}{2} \sum_{i=1}^{N} m_i \mathbf{v}_i^T \mathbf{v}_i - U(\mathbf{x}_1, \cdots, \mathbf{x}_N) \,,$$

with dynamical variables $\mathbf{x} := (\mathbf{x}_1, \cdots, \mathbf{x}_N)^T$ and $\mathbf{v} := (\mathbf{v}_1, \cdots, \mathbf{v}_N)^T$ for the coordinates and velocities of the particles, respectively. In particular, the equation of motion is given by the Euler-Lagrange equation

$$\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \mathbf{v}) = \frac{d}{dt} \nabla_{\mathbf{v}} \mathcal{L}(\mathbf{x}, \mathbf{v}) \,.$$

If we introduce the momenta

$$\mathbf{p}_{\mathbf{x}_i} = \nabla_{\mathbf{v}_i} \mathcal{L}(\mathbf{x}, \mathbf{v}) = m_i \mathbf{v}_i \,, \tag{7}$$

the corresponding Hamiltonian can be written in the form

$$\begin{aligned} \mathcal{H}(\mathbf{x}, \mathbf{p}_{\mathbf{x}}) =& \sum_{i=1}^{N} \frac{\mathbf{p}_{\mathbf{x}_i}^T \mathbf{p}_{\mathbf{x}_i}}{m_i} - \mathcal{L}\left(\mathbf{x}_1, \cdots, \mathbf{x}_N, \frac{\mathbf{p}_{\mathbf{x}_1}}{m_1}, \cdots, \frac{\mathbf{p}_{\mathbf{x}_N}}{m_N}\right) \\ =& \frac{1}{2} \sum_{i=1}^{N} \frac{\mathbf{p}_{\mathbf{x}_i}^T \mathbf{p}_{\mathbf{x}_i}}{m_i} + U(\mathbf{x}_1, \cdots, \mathbf{x}_N) \end{aligned} \tag{8}$$

and Hamilton's equations of motion of the system follow with equation (1).

We now want to follow this way to formulate an isothermal-isobaric (N, P, T) ensemble which allows control of pressure and temperature. The degrees of freedom of an (N, V, E) ensemble, which corresponds to an isolated physical system, are the space coordinates and the velocities of the particles. The idea is to couple the physical system to a fictitious external piston and a fictitious external heat bath; see Figure 1. To this end, additional degrees of freedom for the total coordinate-system in space

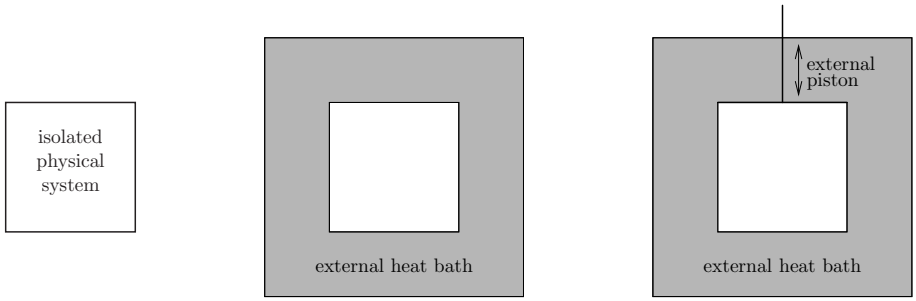


Figure 1: From left to right: The microcanonical or (N, V, E) ensemble, the canonical or (N, V, T) ensemble, and the isothermal-isobaric or (N, P, T) ensemble.

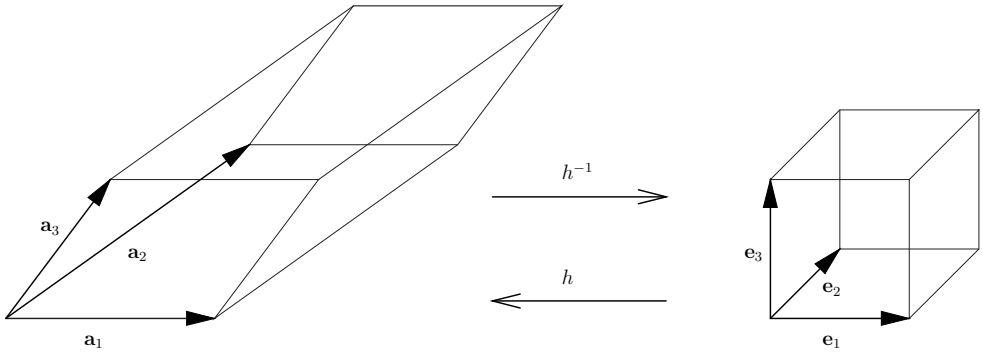


Figure 2: The simulation cell is spanned by the vectors $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$. The vectors $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$ denote the standard orthonormal basis, which span the unit cube in $\mathbb{R}^3$. Here, h denotes the cell transformation matrix.

are introduced. This way, the volume and the shape of the system's domain and thus the pressure can be controlled [42, 50]. Analogously, an additional dynamical variable with respect to rescaling of time is introduced to control the temperature [54, 55, 57].

We assume a simulation cell that is spanned by three linear independent vectors $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$. Now, to obtain dynamical variables to control the volume and the shape of the cell, we combine the basis vectors $\mathbf{a}_1$, $\mathbf{a}_2$ and $\mathbf{a}_3$ into a 3×3 time-dependent cell transformation matrix $\tilde{h} = [\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]$. This way, we can write the coordinate vector of particle i as

$$\mathbf{x}_i = \tilde{h}\tilde{\mathbf{s}}_i \,,$$

where $\tilde{\mathbf{s}}_i$ is the scaled coordinate vector of particle i. In particular, the scaled coordinates $\tilde{\mathbf{s}}_1, \ldots, \tilde{\mathbf{s}}_N$ are located in the unit cube in $\mathbb{R}^3$; see Figure 2. Moreover, the volume of the simulation cell can be calculated as $V = \det \tilde{h}$. To obtain a dynamical variable to control the temperature, we rescale the time t to $\tilde{t}$ by

$$t(\tilde{t}) = \int_0^{\tilde{t}} \frac{d\tau}{\gamma(\tau)} \tag{9}$$

and thus

$$d\tilde{t} = \gamma(\tilde{t})dt \,.$$

This way, the velocities of the particles are scaled according to

$$\mathbf{v}_i(t) = \gamma(\tilde{t})\tilde{h}(\tilde{t})\dot{\tilde{\mathbf{s}}}_i(\tilde{t})\,.$$

Here, the variables that are given in terms of the virtual time $\tilde{t}$ are labeled by a tilde and are the so-called *virtual* variables. Analogously, the variables that are given in terms of the real time t are the so-called *real* variables. Furthermore, we define fictitious potentials of the additional dynamical variables

$$U_P(\tilde{h}) = P_{\text{ext}} \det \tilde{h}\,, \qquad\qquad U_T(\ln\gamma) = N_f k_B T_{\text{tag}} \ln\gamma\,. \tag{10}$$

Here, P_{ext} denotes the externally applied pressure and T_{tag} denotes the target temperature we want to impose onto the system. The fictitious potentials (10) are related to the equation of state (5) for the pressure and the equipartition theorem (4) for the temperature. Now, we extend the Lagrangian of an (N, V, E) ensemble by nine additional degrees of freedom

$$\left\{\tilde{h}_{ij} \;\middle|\; i,j \in \{1,2,3\}\right\}$$

for the so-called Parrinello-Rahman barostat and one degree of freedom γ for the so-called Nosé thermostat. The associated Parrinello-Rahman-Nosé Lagrangian, which corresponds to an (N, P, T) ensemble, is postulated as

$$\begin{aligned}\mathcal{L}_{\text{NPT}}(\tilde{\mathbf{s}},\tilde{h},\gamma,\dot{\tilde{\mathbf{s}}},\dot{\tilde{h}},\dot{\gamma}) =& \frac{1}{2}\sum_{i=1}^{N} m_i\gamma^2\dot{\tilde{\mathbf{s}}}_i^T\dot{\tilde{h}}^T\tilde{h}\dot{\tilde{\mathbf{s}}}_i + \frac{1}{2}M_P\gamma^2\text{tr}\left(\dot{\tilde{h}}^T\dot{\tilde{h}}\right) + \frac{1}{2}M_T\dot{\gamma}^2 \\ & - U(\tilde{h}\tilde{\mathbf{s}},\tilde{h}) - P_{\text{ext}}\det\tilde{h} - N_f k_B T_{\text{tar}}\ln\gamma\,,\end{aligned} \tag{11}$$

where $\text{tr}(\dot{\tilde{h}}^T\dot{\tilde{h}})$ denotes the trace of the matrix $\dot{\tilde{h}}^T\dot{\tilde{h}}$. Here, M_P is a fictitious mass or inertia parameter to control the time-scale of motion of the cell $\tilde{h}$, and M_T is an analogous parameter with respect to temperature. A high value of fictitious mass determines a weak coupling to the external system. In analogy to equation (7), we now have the conjugated momenta

$$\mathbf{p}_{\tilde{\mathbf{s}}_i} = \nabla_{\dot{\tilde{\mathbf{s}}}_i}\mathcal{L}_{\text{NPT}} = m_i\gamma^2\tilde{G}\dot{\tilde{\mathbf{s}}}_i\,, \quad p_{\tilde{h}} = \nabla_{\dot{\tilde{h}}}\mathcal{L}_{\text{NPT}} = \gamma^2 M_P\dot{\tilde{h}}\,, \quad p_\gamma = \nabla_{\dot{\gamma}}\mathcal{L}_{\text{NPT}} = M_T\dot{\gamma}\,.$$

Here, the matrix $\tilde{G}$ is defined as the metric-tensor $\tilde{G} = \tilde{h}^T\tilde{h}$. With equation (8) we obtain the Parrinello-Rahman-Nosé Hamiltonian

$$\begin{aligned}\tilde{\mathcal{H}}_{\text{NPT}}(\tilde{\mathbf{s}},\tilde{h},\gamma,\mathbf{p}_{\tilde{\mathbf{s}}},p_{\tilde{h}},p_\gamma) =& \frac{1}{2}\sum_{i=1}^{N}\frac{\mathbf{p}_{\tilde{\mathbf{s}}_i}^T\tilde{G}^{-1}\mathbf{p}_{\tilde{\mathbf{s}}_i}}{m_i\gamma^2} + \frac{1}{2}\frac{\text{tr}\left(p_{\tilde{h}}^T p_{\tilde{h}}\right)}{\gamma^2 M_P} + \frac{1}{2}\frac{p_\gamma^2}{M_T} \\ & + U(\tilde{h}\tilde{\mathbf{s}},\tilde{h}) + U_P(\tilde{h}) + U_T(\ln\gamma)\,.\end{aligned} \tag{12}$$

In order to derive the equations of motion, we apply equation (1) to the Hamiltonian

(12) and obtain

$$
\begin{aligned}
\frac{d\,\tilde{\mathbf{s}}_i}{d\tilde{t}} &= \nabla_{\mathbf{p}_{\tilde{\mathbf{s}}_i}} \tilde{\mathcal{H}}_{\mathrm{NPT}} = \frac{\tilde{G}^{-1}\mathbf{p}_{\tilde{\mathbf{s}}_i}}{m_i\gamma^2}\,, \\
\frac{d\,\mathbf{p}_{\tilde{\mathbf{s}}_i}}{d\tilde{t}} &= -\nabla_{\tilde{\mathbf{s}}_i} \tilde{\mathcal{H}}_{\mathrm{NPT}} = -\nabla_{\tilde{\mathbf{s}}_i} \tilde{U}(\tilde{\mathbf{s}},\tilde{h})\,, \\
\frac{d\,\tilde{h}}{d\tilde{t}} &= \nabla_{p_{\tilde{h}}} \tilde{\mathcal{H}}_{\mathrm{NPT}} = \frac{p_{\tilde{h}}}{M_P\gamma^2}\,, \\
\frac{d\,\mathbf{p}_{\tilde{h}}}{d\tilde{t}} &= -\nabla_{\tilde{h}} \tilde{\mathcal{H}}_{\mathrm{NPT}} = -\nabla_{\tilde{h}} \left(\frac{1}{2}\sum_{i=1}^{N} \frac{\mathbf{p}_{\tilde{\mathbf{s}}}^T \tilde{G}^{-1}\mathbf{p}_{\tilde{\mathbf{s}}}}{m_i\gamma^2} \right) - \nabla_{\tilde{h}} \tilde{U}(\tilde{\mathbf{s}},\tilde{h}) - \nabla_{\tilde{h}} U_P(\tilde{h}) \\
\frac{d\,\gamma}{d\tilde{t}} &= \nabla_{p_\gamma} \tilde{\mathcal{H}}_{\mathrm{NPT}} = \frac{p_\gamma}{M_T}\,, \\
\frac{d\,p_\gamma}{d\tilde{t}} &= -\nabla_{\gamma} \tilde{\mathcal{H}}_{\mathrm{NPT}} = \frac{1}{\gamma}\left(\sum_{i=1}^{N} \frac{\mathbf{p}_{\tilde{\mathbf{s}}_i}^T \tilde{G}^{-1}\mathbf{p}_{\tilde{\mathbf{s}}_i}}{m_i\gamma^2} + \frac{\mathrm{tr}\left(p_{\tilde{h}}^T p_{\tilde{h}}\right)}{m_i\gamma^2} - N_f k_B T_{\mathrm{tag}} \right)\,,
\end{aligned}
\tag{13}
$$

where $\tilde{U}$ is defined as

$$
\tilde{U}(\tilde{\mathbf{s}},\tilde{h}) := U(\tilde{h}\tilde{\mathbf{s}},\tilde{h})\,.
$$

The equations of motion (13) could be solved numerically. To this end, an appropriate time discretization scheme has to be applied. But the Hamiltonian (12) and the equations of motion (13) are given in term of the virtual time $\tilde{t}$. Therefore, the use of a constant time-step for the scaled or virtual time leads to a non-constant time-step of the non-scaled real time. In order to avoid the difficulties that arise for a time discretization scheme with varying time steps, we transform the equations of motion (13) back to real time [55, 57, 58, 65]. To this end, we define new variables by

$$
\mathbf{s}_i(t) := \tilde{\mathbf{s}}_i(\tilde{t})\,, \qquad\qquad h(t) := \tilde{h}(\tilde{t})\,. \tag{14}
$$

Their derivative with respect to time reads as

$$
\frac{d\,\mathbf{s}_i}{dt}(t) = \gamma(\tilde{t})\frac{d\,\tilde{\mathbf{s}}_i}{d\tilde{t}}(\tilde{t})\,, \qquad\qquad \frac{d\,h}{dt}(t) = \gamma(\tilde{t})\frac{d\,\tilde{h}}{d\tilde{t}}(\tilde{t})\,, \tag{15}
$$

compare also equation (9). We transform the conjugated momenta accordingly by

$$
\mathbf{p}_{\mathbf{s}_i} := \frac{\tilde{G}^{-1}\mathbf{p}_{\tilde{\mathbf{s}}_i}}{\gamma}\,, \qquad\qquad p_h := \frac{p_{\tilde{h}}}{\gamma}\,, \tag{16}
$$

and obtain for the time derivatives

$$
\begin{aligned}
\frac{d\,\mathbf{p}_{\mathbf{s}_i}}{dt}(t) &= \frac{d\left(\tilde{G}^{-1}\mathbf{p}_{\tilde{\mathbf{s}}_i}\frac{1}{\gamma}\right)}{d\tilde{t}}(\tilde{t}) = \tilde{G}^{-1}(\tilde{t})\left(\frac{d\,\tilde{G}}{d\tilde{t}}(\tilde{t})\tilde{G}^{-1}(\tilde{t}) + \frac{d\,\mathbf{p}_{\tilde{\mathbf{s}}_i}}{d\tilde{t}}(\tilde{t}) - \frac{\mathbf{p}_{\tilde{\mathbf{s}}_i}(\tilde{t})}{\gamma(\tilde{t})} \right)\,, \\
\frac{d\,p_h}{dt}(t) &= \gamma(\tilde{t})\frac{d\left(p_{\tilde{h}}\frac{1}{\gamma}\right)}{d\tilde{t}}(\tilde{t}) = \frac{d\,p_{\tilde{h}}}{d\tilde{t}}(\tilde{t}) - \frac{p_{\tilde{h}}(\tilde{t})}{\gamma(\tilde{t})}\frac{d\,\gamma}{d\tilde{t}}(\tilde{t})\,.
\end{aligned}
\tag{17}
$$

Note that we multiplied the conjugated momenta of the particles by the inverse metric-tensor to express the equations of motion in simple terms. For the same reason, we took the logarithm of the variable for the temperature control

$$
\eta(t) := \ln\gamma(\tilde{t})\,, \qquad\qquad p_\eta(t) = p_\gamma(\tilde{t})\,. \tag{18}
$$

We thus obtain

$$\frac{d\,\eta}{dt}(t) = \gamma(\tilde{t})\frac{d}{d\tilde{t}}\ln\gamma(\tilde{t}) = \frac{d\,\gamma}{d\tilde{t}}(\tilde{t})\,, \qquad \frac{d\,p_\eta}{dt}(t) = \frac{d\,p_\gamma}{d\tilde{t}}(\tilde{t}) \tag{19}$$

for the derivatives with respect to time. Finally, we use the equations of motion (13) and the equations (14), (16), (18), together with $G(t) := \tilde{G}(\tilde{t})$, to express equations (15), (17), and (19) in term of real time only, and obtain the transformed equations of motion:

$$\dot{\mathbf{s}}_i = \frac{\mathbf{p}_{\mathbf{s}_i}}{m_i}\,, \quad \dot{h} = \frac{p_h}{M_P}\,, \quad \dot{\eta} = \frac{p_\eta}{M_T}\,, \tag{20}$$

$$\dot{\mathbf{p}}_{\mathbf{s}_i} = -h^{-1}\nabla_{\mathbf{x}_i}U - G^{-1}\dot{G}p_{\mathbf{s}_i} - \frac{p_\eta}{M_T}p_{\mathbf{s}_i}\,, \tag{21}$$

$$\dot{p}_h = -\sum_{i=1}^{N}\nabla_{\mathbf{x}_i}U\mathbf{s}_i^T - \nabla_h U + \sum_{i=1}^{N} m_i h\dot{\mathbf{s}}_i\dot{\mathbf{s}}_i^T - h^{-T}P_{\text{ext}}\det h - \frac{p_\eta}{M_T}p_h\,, \tag{22}$$

$$\dot{p}_\eta = \sum_{i=1}^{N}\frac{\mathbf{p}_{\mathbf{s}_i}^T G\mathbf{p}_{\mathbf{s}_i}}{m_i} + \frac{\operatorname{tr}(p_h^T p_h)}{M_P} - N_f k_B T\,. \tag{23}$$

In the same way, we can transform the Hamiltonian (12) to express it in term of real time only. The resulting function can be written as

$$\begin{aligned}\mathcal{H}_{\text{NPT}}(\mathbf{s}, h, \eta, \mathbf{p}_\mathbf{s}, p_h, p_\eta) =& \frac{1}{2}\sum_{i=1}^{N}\frac{\mathbf{p}_{\mathbf{s}_i}^T G\mathbf{p}_{\mathbf{s}_i}}{m_i} + \frac{1}{2}\frac{\operatorname{tr}(p_h^T p_h)}{M_P} + \frac{1}{2}\frac{p_\eta^2}{M_T} \\ &+ U(h\mathbf{s}, h) + P_{\text{ext}}\det h + N_f k_B T_{\text{tag}}\eta\,.\end{aligned} \tag{24}$$

Like the Hamiltonian (2) of an (N, V, E) ensemble, the function $\mathcal{H}_{\text{NPT}}$ remains constant over time. But the non-fictitious physical energy term $E_{\text{kin}} + E_{\text{pot}}$ is not conserved over time. Here, the physical kinetic energy reads as $E_{\text{kin}} = \frac{1}{2}\sum_{i=1}^{N}\frac{\mathbf{p}_{\mathbf{s}_i}^T G\mathbf{p}_{\mathbf{s}_i}}{m_i}$ and the physical potential energy is given by $E_{\text{pot}} = U(h\mathbf{s}, h)$. Note that the function $\mathcal{H}_{\text{NPT}}$ is not a Hamiltonian, because the equations of motion (20)–(23) can not be derived from it. Thus, the function $\mathcal{H}_{\text{NPT}}$ together with the equations of motion (20)–(23) correspond to so-called *non-Hamiltonian* dynamics [70]. However, analogous to equation (4), the thermodynamic temperature T is given as the time average of the instantaneous temperature

$$T = \langle T_{\text{inst}}\rangle_t\,, \qquad T_{\text{inst}} = \frac{2E_{\text{kin}}}{N_f k_B}\,, \qquad E_{\text{kin}} = \frac{1}{2}\sum_{i=1}^{N}\frac{\mathbf{p}_{\mathbf{s}_i}^T G\mathbf{p}_{\mathbf{s}_i}}{m_i}\,.$$

The instantaneous pressure given in equation (6) now depends on the fluctuations of the potential energy U with respect to the volume V. In particular, it can be written in the form

$$P_{\text{inst}} = \frac{2}{3V}E_{\text{kin}} + \frac{1}{V}W_{\text{inst}} = \frac{2}{3V}E_{\text{kin}} - \frac{dU}{dV}\,.$$

With the help of the relation $V = \det h$ and the chain rule, P_{inst} can also be obtained from the so-called instantaneous internal stress tensor Π_{int} by $P_{\text{int}} = \frac{1}{3}\operatorname{tr}(\Pi_{\text{int}})$. Here,

Π_{int} is defined as

$$\Pi_{\text{int}} = \frac{1}{\det h} \sum_{i=1}^{N} m_i h \mathbf{s}_i \mathbf{s}_i^T h^T + \Pi_{\text{int}}^{\text{pot}} , \quad \Pi_{\text{int}}^{\text{pot}} = \frac{1}{\det h} F_h h^T , \\ (F_h)_{\alpha\beta} := -\frac{d}{d\, h_{\alpha\beta}} U , \quad \alpha, \beta \in \{1, 2, 3\} . \tag{25}$$

We can express the equation of motion (22) in terms of the instantaneous internal stress tensor as

$$\dot{p}_h = (\Pi_{\text{int}} - P_{\text{ext}} I)\, h^{-T} \det h - \frac{p_\eta}{M_T} p_h ,$$

where I denotes the 3×3 identity matrix. If we now assume that the potential U does not depend explicitly on the cell matrix h, the 3×3 matrix

$$F_h = -\frac{d}{d\, h} U(h\mathbf{s}_1, \ldots, h\mathbf{s}_N) \tag{26}$$

can be written as

$$F_h = \sum_{i=1}^{N} \mathbf{F}_i \mathbf{s}_i^T \tag{27}$$

by exploiting the chain rule. Here, $\mathbf{F}_i$ denotes the force that is contributed by the potential U and acts on particle i. Analogous to equation (3) it reads as $\mathbf{F}_i = -\nabla_{\mathbf{x}_i} U$. To perform molecular dynamics simulations for nanocomposites, we usually apply periodic boundary conditions. In that case, however, the potential is explicitly dependent on the unit cell matrix h. We will discuss periodic potentials as well as the related derivation of the term F_h in detail in the next section.

From a formal point of view, it is desirable to keep the Hamiltonian structure of the equations of motion (13) which is however lost due to the noncanonical transformation back to real time given by the definitions (14), (16), and (18) [70, 71]. To overcome this drawback, there exist some variants of the presented Parrinello-Rahman-Nosé approach [71]. For example, in the so-called Nosé-Poincaré method for constant temperature molecular dynamics, a Poincaré transformation, which is in particular a canonical transformation, is used to obtain a Hamiltonian system that provides trajectories at evenly spaced points in real time [72].

With regard to constant pressure molecular dynamics, the Parrinello-Rahman formulation exhibits two difficulties. First, the choice of the artificial kinetic energy term in equation (11) leads to unphysical symmetry breaking effects, because the equations of motion are now not invariant under so-called modular transformations [73, 74]. Second, three of the nine dynamical variables for the control of the size and the shape of the simulation cell are redundant, as the absolute orientation of the simulation cell is physically meaningless [54]. In the literature, several methods can be found to get rid of nonphysically simulation cell rotations [54, 75, 76, 77]. In particular, the choice of the six independent degrees of freedom of the metric tensor G as dynamical variables to control pressure leads to a method where the artificial kinetic energy term is invariant under modular transformations and for which nonphysical cell rotations do not appear [78]. Note that this metric-tensor based approach and the Nosé-Poincaré method have been combined to obtain a Hamiltonian system that

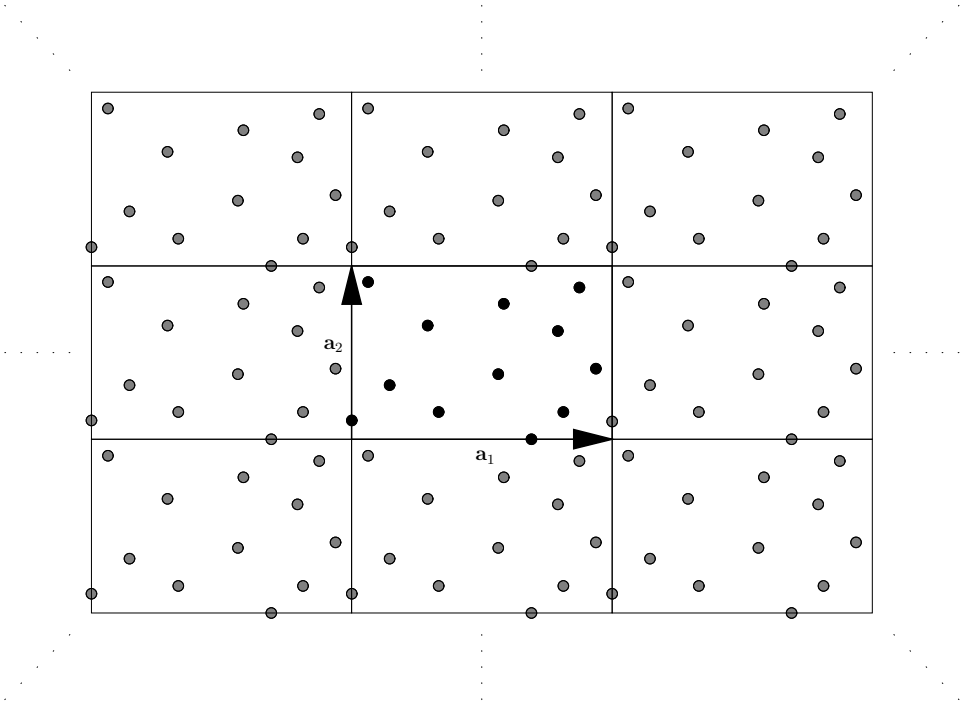


Figure 3: Periodically translated images of the simulation unit cell in two dimensions. The black dots are particles within the unit cell and the gray dots are their periodic images. To obtain uniqueness at the cell boundaries, see convention (28).

is related to an isothermal-isobaric ensemble [71]. As we will later discuss in section 2.3, we follow a different approach and simply constrain the matrix $\dot{p}_h$ to be symmetric to avoid a nonphysical behavior of the simulation cell.

2.1.2 Periodic Boundary Conditions

In a molecular dynamics simulation for nanocomposites, it is natural to apply periodic boundary conditions [79]. This way, an N-particle system in a simulation cell of finite volume V is replicated infinitely throughout space by translations; see also Figure 3. In particular, the ratio of the total number of particles and the total volume remains constant for the induced infinite system. If periodic boundary conditions are used, atoms in the unit cell interact not only with other atoms in the same unit cell but also with their translated images. Here, to obtain uniqueness at the two dimensional boundaries of the cell, we use the convention

$$\mathbf{s}_1, \ldots, \mathbf{s}_N \in [0,1)^3 \,. \tag{28}$$

To include the interactions of the particles with the translated images of the particles correctly, a periodic potential must be used, which usually depends explicitly on the cell matrix h. This dependence on h has to be carefully taken into account in the evaluation of the potential contribution to the expressions $\Pi_{\text{int}}^{\text{pot}}$ and $\mathbf{F}_i$. Note that a periodic potential gives the potential energy per unit cell. We discuss in this section how to apply periodic boundary conditions for two-body potentials of short range

as well as for bond potentials with static bond relations, which will later be used to model hydrocarbon systems. Furthermore, we briefly discuss the treatment of angle- and torsional-dependent potential terms.

First we consider a potential

$$U_{\text{two}}(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j>i}}^{N} u_{\text{two}}^{ij}(r_{ij}) \,,$$

which only involves two-body terms $u_{\text{two}}^{ij}(r_{ij})$ with $u_{\text{two}}^{ij} \equiv u_{\text{two}}^{ji}$. Here, r_{ij} denotes the distance between particle i and j, which is given by the norm of the distance vector $r_{ij} := \|\mathbf{r}_{ij}\|$. Furthermore, we define the distance vector $\mathbf{r}_{ij}$ in the form $\mathbf{r}_{ij} := \mathbf{x}_j - \mathbf{x}_i$. To deal with the periodic setting, we index a translated image of a particle i by $i_{\mathbf{S}}$, with $\mathbf{S} \in \mathbb{Z}^3$. The coordinates of a translated image of a particle i are then written with the help of a translation vector $h\mathbf{S}$ in the form $\mathbf{x}_{i_{\mathbf{S}}} := \mathbf{x}_i + h\mathbf{S}$. Thus, the distance vector $\mathbf{r}_{ij_{\mathbf{S}}}$ and the distance $r_{ij_{\mathbf{S}}}$ between a particle i and a translated image of a particle j read as

$$\mathbf{r}_{ij_{\mathbf{S}}} = \mathbf{x}_{j_{\mathbf{S}}} - \mathbf{x}_i = \mathbf{x}_j + h\mathbf{S} - \mathbf{x}_i \,, \qquad\qquad r_{ij_{\mathbf{S}}} = \|\mathbf{r}_{ij_{\mathbf{S}}}\| \,.$$

This way, the periodic potential $U_{\text{two}}^{\text{periodic}}$ which is associated to the potential U_{two} can be written as

$$U_{\text{two}}^{\text{periodic}}(\mathbf{x}_1, \ldots, \mathbf{x}_N, h) = \sum_{i=1}^{N} \left(\sum_{\substack{j=1 \\ j>i}}^{N} u_{\text{two}}^{ij}(r_{ij}) + \sum_{j=1}^{N} \sum_{\mathbf{S} \in \mathcal{S}} u_{\text{two}}^{ij}(r_{ij_{\mathbf{S}}}) \right) , \tag{29}$$

where $\mathbf{S}$ is in the set $\mathcal{S} := \mathbb{Z}^3 \backslash \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$. We first consider two-body potential terms of short range with

$$u_{\text{two}}^{ij}(r) = 0 \,, \qquad\qquad \text{for all } r > r_c \,, \tag{30}$$

where r_c is positive; see Figure 4. Thus, only a finite number of terms in the infinite sum of equation (29) are nonzero. For reasons of simplicity, we further assume that

$$0 < 2r_c < \|\mathbf{a}_j^{\perp}\| = \frac{1}{\|\mathbf{b}_j\|} \,, \quad \text{with } \mathbf{a}_j^{\perp} := \left(\frac{\mathbf{b}_j^T}{\|\mathbf{b}_j\|} \mathbf{a}_j \right) \frac{\mathbf{b}_j}{\|\mathbf{b}_j\|} \,, \quad \text{for } j \in \{1,2,3\} \,, \tag{31}$$

where the so-called reciprocal basis vectors $\mathbf{b}_j$ are defined by

$$[\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3] := h^{-T} \,. \tag{32}$$

and thus $\mathbf{b}_j^T \mathbf{a}_i = \delta_{ij}$ for $i, j \in \{1,2,3\}$. Here, δ_{ij} denotes the Kronecker delta with

$$\delta_{ij} = \begin{cases} 0 & \text{if } i = j \,, \\ 1 & \text{if } i \neq j \,. \end{cases}$$

In equation (31), each vector $\mathbf{a}_i^{\perp}$ is defined as the orthogonal projection of $\mathbf{a}_i$ onto the linear span of the subset of the normalized reciprocal basis vectors given by

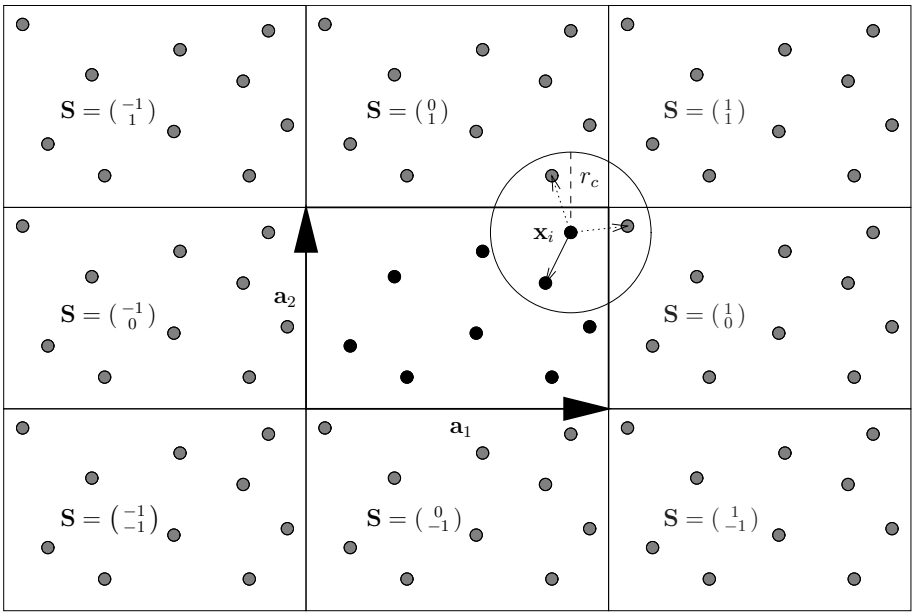


Figure 4: A two-body potential term of short range with periodic boundary conditions. Distance vectors between a particle $\mathbf{x}_i$ in the simulation unit cell and a particle in a translated simulation cell are indicated by a dotted arrow. The translation vectors are given by $h\mathbf{S}$ with $h = [\mathbf{a}_1, \mathbf{a}_2]$ in the two-dimensional case.

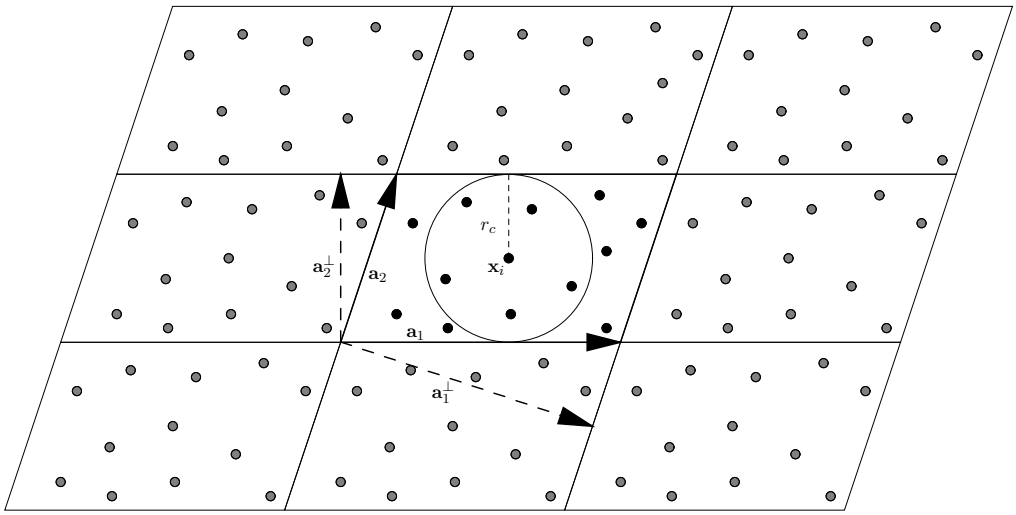


Figure 5: A sheared simulation cell in two dimensions with periodic boundary conditions; see also equation (31) and (32).

$\left\{\frac{\mathbf{b}_j}{\|\mathbf{b}_j\|} : j \neq i\,, j \in \{1,2,3\}\right\}$. This way, also in the case of a sheared simulation cell, there is no interaction sphere for a particle $\mathbf{x}_i$ that contains a particle and any of its translated images; see Figure 5. With the two-body forces $\mathbf{F}_{ij} = -\frac{d}{d\mathbf{x}_i} u_{\text{two}}(r_{ij})$, the potential contribution to the force on the particle i in equation (21) can be written in the form

$$\mathbf{F}_i = -\nabla_{\mathbf{x}_i} U_{\text{two}}^{\text{periodic}} = \sum_{\substack{j=1 \\ j\neq i}}^{N} \mathbf{F}_{ij} + \sum_{j=1}^{N} \sum_{\mathbf{S}\in\mathcal{S}} \mathbf{F}_{ij_\mathbf{S}} \tag{33}$$

$$= \sum_{\substack{j=1 \\ j\neq i}}^{N} \sum_{\mathbf{S}\in\mathcal{S}'} \mathbf{F}_{ij_\mathbf{S}}\,, \tag{34}$$

with $\mathcal{S}' := \left\{\begin{pmatrix} i \\ j \\ k \end{pmatrix} : i,j,k \in \{-1,0,1\}\right\}$. Note that for the last equality we make use of the assumptions (30) and (31). Furthermore, the symmetry of the two-body terms $u_{\text{two}}^{ij}(r_{ij}) = u_{\text{two}}^{ji}(r_{ji})$ leads to Newton's third law

$$\mathbf{F}_{ij} = -\mathbf{F}_{ji}\,, \tag{35}$$

and the corresponding relation

$$\mathbf{F}_{ij}\mathbf{x}_i^T + \mathbf{F}_{ji}\mathbf{x}_j^T = -\mathbf{F}_{ij}\mathbf{r}_{ij}^T\,. \tag{36}$$

To write the potential contribution of the internal stress tensor in equation (25) in terms of two-body forces and distances, we apply the chain rule similar to equations (26) and (27). We then use equations (35) and (36) and obtain

$$\Pi_{\text{int}}^{\text{pot}} = \frac{1}{\det h} \sum_{i=1}^{N} \left(\sum_{\substack{j=1 \\ j>i}}^{N} -\mathbf{F}_{ij}\mathbf{r}_{ij}^T + \sum_{\substack{j=1 \\ j>i}}^{N} \sum_{\mathbf{S}\in\mathcal{S}} -\mathbf{F}_{ij_\mathbf{S}}\mathbf{r}_{ij_\mathbf{S}}^T \right) \tag{37}$$

$$= \frac{1}{\det h} \sum_{i=1}^{N} \left(\sum_{\substack{j=1 \\ j>i}}^{N} \sum_{\mathbf{S}\in\mathcal{S}'} -\mathbf{F}_{ij_\mathbf{S}}\mathbf{r}_{ij_\mathbf{S}}^T \right)\,. \tag{38}$$

For the second equality we again make use of the assumptions (30) and (31). Note that the expression of the potential contribution of the internal stress tensor given in (27) is not sufficient for the periodic boundary case, because the contribution of the translated images is not correctly included.

Now, we consider two, three, and four body potentials with static bond relations. Note that these potential models are used to describe intramolecular forces; see also Figure 6. Note further that for a wide range of molecules the necessary parameters for these potentials can be obtained from databases like CHARMM [80] or AMBER [81]. Now, a static bond model that involves only bond potential terms $u_{\text{bond}}^{ij}(r_{ij})$ with $u_{\text{bond}}^{ij} \equiv u_{\text{bond}}^{ji}$ is defined by

$$U_{\text{bond}}(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \sum_{(i,j)\in\mathcal{B}} u_{\text{bond}}^{ij}(r_{ij})\,.$$

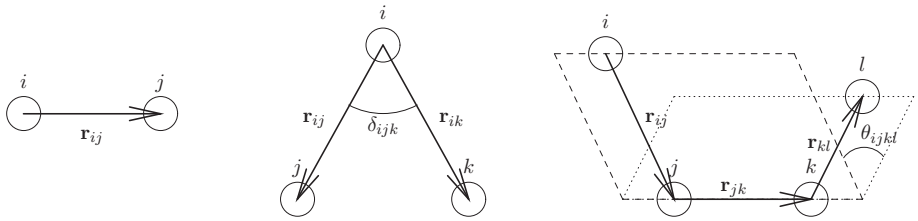


Figure 6: Schematic diagram of bond stretching (left), bond angle bending (center), and torsional (right) potentials to model intramolecular interactions.

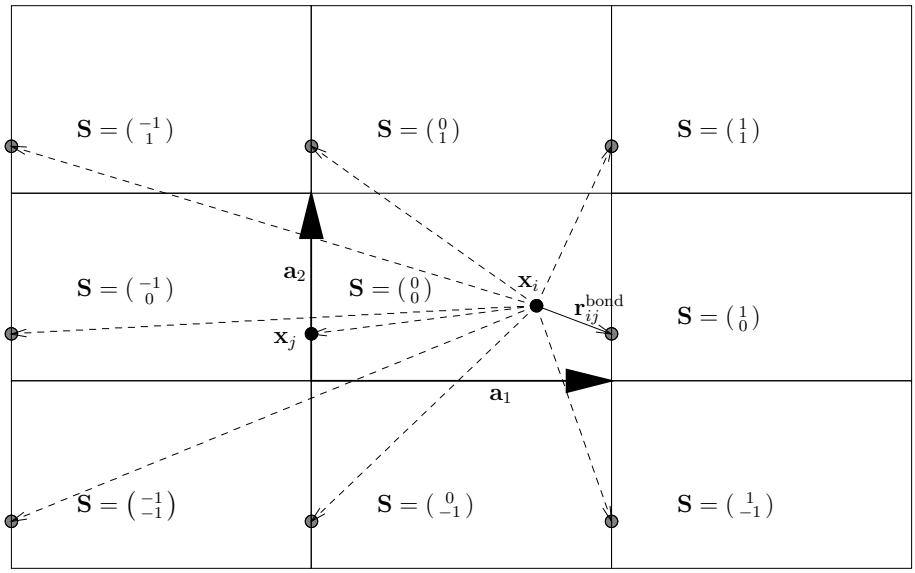


Figure 7: A bond potential term in two dimensions with periodic boundary conditions. Here, the bond distance vector $\mathbf{r}_{ij}^{\text{bond}}$ is given by the distance vector $\mathbf{r}_{ij_{\binom{1}{0}}}$; see also equation (39).

Here, we assume that the set of bonds $\mathcal{B}$ is a subset of the set of all pairs $\{(i,j) \in \{1,\dots,N\}^2 \,:\, i<j\}$. In contrast to the two-body potential given in equation (29), we no longer have to consider each of the pairs $\left\{(i,j_{\mathbf{S}}) \,:\, \mathbf{S}\in\mathbb{Z}^3\right\}$ for a given particle pair (i,j). Instead, in the case of a bond potential, we only have to choose one of the pairs $\left\{(i,j_{\mathbf{S}}) \,:\, \mathbf{S}\in\mathbb{Z}^3\right\}$ for a given bond $(i,j)\in\mathcal{B}$. To this end, we define for each bond $(i,j)\in\mathcal{B}$ a distance vector $\mathbf{r}_{ij_{\mathbf{S}}}$ with $\mathbf{S}\in\mathcal{S}$, which minimizes the distance $r_{ij_{\mathbf{S}}}$. In particular, due to the convention (28) and the exclusion of bond pairs (i,i), the bond distance vector

$$\mathbf{r}_{ij}^{\text{bond}} \in \left\{ \mathbf{r}_{ij_{\mathbf{S}}} \,:\, \|\mathbf{r}_{ij_{\mathbf{S}}}\| = \min_{\mathbf{S}\in\mathbb{Z}^3} \{r_{ij_{\mathbf{S}}}\} \right\} \tag{39}$$

is unique for each bond pair (i,j); see Figure 7. Thus, the corresponding periodic potential can be defined in the form

$$U_{\text{bond}}^{\text{periodic}}(\mathbf{x}_1,\dots,\mathbf{x}_N,h) = \sum_{(i,j)\in\mathcal{B}} u_{\text{bond}}^{ij}(r_{ij}^{\text{bond}})\,. \tag{40}$$

The potential contribution to the forces on the particles in equation (21) can then be written in the form

$$\mathbf{F}_k = -\nabla_{\mathbf{x}_k} U_{\text{bond}}^{\text{periodic}} = \sum_{\substack{(i,j)\in\mathcal{B} \\ i=k \vee j=k}} -\frac{d}{d\mathbf{x}_k} u_{\text{bond}}(r_{ij}^{\text{bond}}) \,. \tag{41}$$

Analogously to the two-body potential case in equation (37), the potential contribution to the internal stress tensor in equation (25) reads with the bond forces $\mathbf{F}_{ij} = -\frac{d}{d\mathbf{x}_i} u_{\text{bond}}(r_{ij}^{\text{bond}})$ as

$$\Pi_{\text{int}}^{\text{pot}} = \frac{1}{\det h} \sum_{(i,j)\in\mathcal{B}} -\mathbf{F}_{ij} \left(\mathbf{r}_{ij}^{\text{bond}}\right)^T \,. \tag{42}$$

In a way similar to two-body potential terms u_{two} and bond potential terms u_{bond}, we treat angle-dependent potential terms $u_{\text{ang}}(\theta_{ijk})$ and torsional-dependent terms $u_{\text{tor}}(\phi_{ijkl})$. Here, θ_{ijk} denotes the angle between $\mathbf{r}_{ij}$ and $\mathbf{r}_{ik}$ and ϕ_{ijkl} denotes the so-called torsional or dihedral angle $\sphericalangle(\mathbf{r}_{ij} \times \mathbf{r}_{jk}, \mathbf{r}_{jk} \times \mathbf{r}_{kl})$. The corresponding periodic potential can be defined in analogously to equation (40). For an angle-dependent potential term, we define the force contribution in the form $\mathbf{F}_n^{ijk} := -\frac{d}{d\mathbf{x}_n} u_{\text{ang}}(\theta_{ijk})$. Then, relations similar Newton's third law (35) and equation (36) hold with

$$\begin{aligned} \mathbf{F}_i^{ijk} &= -\mathbf{F}_j^{ijk} - \mathbf{F}_k^{ijk} \,, \\ \mathbf{F}_i^{ijk}\mathbf{r}_i + \mathbf{F}_j^{ijk}\mathbf{r}_j + \mathbf{F}_k^{ijk}\mathbf{r}_k &= \mathbf{F}_j^{ijk}\mathbf{r}_{ij}^T + \mathbf{F}_k^{ijk}\mathbf{r}_{ik}^T \,. \end{aligned} \tag{43}$$

In the torsional-dependent case, relations similar to Newton's third law (35) and equation (36) can be expressed with the help of potential force contributions $\mathbf{F}_n^{ijkl} := -\frac{d}{d\mathbf{x}_n} u_{\text{tor}}(\phi_{ijkl})$ as follows:

$$\begin{aligned} \mathbf{F}_i^{ijkl} &= -\mathbf{F}_j^{ijkl} - \mathbf{F}_k^{ijkl} - \mathbf{F}_l^{ijkl} \,, \\ \mathbf{F}_i^{ijkl}\mathbf{r}_i + \mathbf{F}_j^{ijkl}\mathbf{r}_j + \mathbf{F}_k^{ijkl}\mathbf{r}_k + \mathbf{F}_l^{ijkl}\mathbf{r}_l &= -\mathbf{F}_i^{ijkl}\mathbf{r}_{ij}^T - (\mathbf{F}_i^{ijkl} + \mathbf{F}_j^{ijkl})\mathbf{r}_{jk}^T + \mathbf{F}_l^{ijkl}\mathbf{r}_{kl}^T \,. \end{aligned} \tag{44}$$

Like in the case of two-body and bond potentials, the equations (43) and (44) can be exploited to express the potential contribution expressions $\Pi_{\text{int}}^{\text{pot}}$ and $\mathbf{F}_i$ only in terms of distance vectors and force terms $\mathbf{F}_n^{ijk}$ and $\mathbf{F}_n^{ijkl}$. This is helpful for an efficient implementation as we will discuss later. Note that the contribution of the torsional-dependent potential to the trace of the stress tensor is zero and therefore the contribution to the pressure is zero as well [82]. However, the contribution to the full stress tensor must be taken into account. Note furthermore that the internal stress tensor Π_{int} is symmetric in the case of pair, angle-dependent and torsional-dependent potentials.

2.1.3 Time Integration

For the numerical solution of the equations of motion (20)-(23), we have to employ a time integration scheme. At first, we consider a Hamiltonian system with separable Hamiltonian

$$\mathcal{H}(\mathbf{q},\mathbf{p}) = K(\mathbf{p}) + U(\mathbf{q}) = \frac{1}{2}\mathbf{p}^T m^{-1}\mathbf{p} + U(\mathbf{q}) \,.$$

Then, the corresponding Hamilton's equations of motion are

$$\dot{\mathbf{q}} = \nabla_{\mathbf{p}} \mathcal{H}(\mathbf{q}, \mathbf{p}) = m^{-1}\mathbf{p}\,, \qquad \dot{\mathbf{p}} = \mathbf{F} = -\nabla_{\mathbf{q}} \mathcal{H}(\mathbf{q}, \mathbf{p}) = -\nabla_{\mathbf{q}} U(\mathbf{q})\,, \tag{45}$$

with generalized coordinates $\mathbf{q}$, momenta $\mathbf{p}$ and an invertible diagonal mass matrix m [83]; see also equations (2) and (1). For the numerical solution of system (45) of first-order ordinary differential equations, we have to discretize in time. Thus, we consider the system (45) only at time points, $t_n := t_0 + n\Delta t$ with $n \in \{0, 1, \ldots, n_{\max}\}$. By a Taylor expansion up to third order of the coordinates $\mathbf{q}(t + \Delta t)$ and $\mathbf{q}(t - \Delta t)$ around time t, we obtain

$$\mathbf{q}(t + \Delta t) = \mathbf{q}(t) + \Delta t m^{-1}\mathbf{p}(t) + \frac{1}{2}\Delta t^2 m^{-1}\mathbf{F}(t) + \frac{1}{6}\Delta t^3 \frac{d^3\mathbf{q}(t)}{dt^3} + \mathcal{O}(\Delta t^4) \tag{46}$$

and

$$\mathbf{q}(t - \Delta t) = \mathbf{q}(t) - \Delta t m^{-1}\mathbf{p}(t) + \frac{1}{2}\Delta t^2 m^{-1}\mathbf{F}(t) - \frac{1}{6}\Delta t^3 \frac{d^3\mathbf{q}(t)}{dt^3} + \mathcal{O}(\Delta t^4)\,, \tag{47}$$

under appropriate smoothness assumptions on $\mathbf{q}$. The sum of equations (46) and equation (47) gives

$$\mathbf{q}(t + \Delta t) + \mathbf{q}(t - \Delta t) = 2\mathbf{q}(t) + \Delta t^2 m^{-1}\mathbf{F}(t) + \mathcal{O}(\Delta t^4) \tag{48}$$

and the difference results in

$$\mathbf{q}(t + \Delta t) - \mathbf{q}(t - \Delta t) = 2\Delta t m^{-1}\mathbf{p}(t) + \mathcal{O}(\Delta t^3)\,. \tag{49}$$

Equation (48) leads to an approximation for the coordinates

$$\mathbf{q}(t + \Delta t) \approx 2\mathbf{q}(t) - \mathbf{q}(t - \Delta t) + \Delta t^2 m^{-1}\mathbf{F}(t)\,, \tag{50}$$

with an error of order $\mathcal{O}(\Delta t^4)$. Equation (49) leads with help of equation (46) to an approximation for the momenta

$$\mathbf{p}(t + \Delta t) \approx \mathbf{p}(t) + \frac{\Delta t}{2}\left(\mathbf{F}(t) + \mathbf{F}(t + \Delta t)\right)\,, \tag{51}$$

with an error of order $\mathcal{O}(\Delta t^2)$. Equation (50) and equation (51) correspond to the so-called velocity-Störmer-Verlet time-integration scheme [84]:

$$\begin{aligned} \mathbf{q}^0 &:= \mathbf{q}(t_0)\,, \quad \mathbf{p}^0 := \mathbf{p}(t_0)\,, \\ \mathbf{q}^{n+1} &:= 2\mathbf{q}^n - \Delta t \mathbf{q}^{n-1} + \Delta t^2 m^{-1}\mathbf{F}^n\,, \qquad (52) \\ \mathbf{p}^{n+1} &:= \mathbf{p}^n + \frac{\Delta t}{2}\left(\mathbf{F}^n + \mathbf{F}^{n+1}\right)\,. \qquad (53) \end{aligned}$$

There are two more formulations of the velocity-Störmer-Verlet algorithm, the original Störmer-Verlet scheme [85, 86] and the so-called Leapfrog scheme [87]. With exact arithmetic, these time-integration schemes are all equivalent. Because the momenta $\mathbf{p}$ are not used in equation (52), their comparatively low accuracy of the order $\mathcal{O}(\Delta t^2)$ has no impact on the accuracy of the coordinates $\mathbf{q}$. It however affects the calculation of the temperature and the kinetic energy. To improve the accuracy for

these quantities, the so-called Beeman algorithm can be employed [88]. The third-order Beeman scheme is also derived from Taylor expansions around time t_n up to third order. It is given by

$$\mathbf{q}^{n+1} := \mathbf{q}^n + \Delta t m^{-1}\mathbf{p}^n + \frac{\Delta t^2}{6} m^{-1}\left(4\mathbf{F}^n - \mathbf{F}^{n-1}\right) , \tag{54}$$

$$\mathbf{p}^{n+1} := \mathbf{p}^n + \frac{\Delta t}{6}\left(2\mathbf{F}^{n+1} + 5\mathbf{F}^n - \mathbf{F}^{n-1}\right) . \tag{55}$$

If we use equation (55) to eliminate the momenta $\mathbf{p}^n$ in equation (54), we can see easily that the Beeman scheme yields the same trajectories $\mathbf{q}^n$ as the Störmer-Verlet based algorithms. But now, the error of the momenta is of order $\mathcal{O}(\Delta t^3)$ instead of order $\mathcal{O}(\Delta t^2)$, and thus the total energy conservation is better by one order than for the Störmer-Verlet case [65].

Now, we consider the equations of motions (20)-(23), which are related to an (N, P, T) ensemble. Note that these equations in particular correspond to a non-Hamiltonian dynamics. With coordinates $\mathbf{q} = (\mathbf{s}, h, \eta)$ and momenta $\mathbf{p} := (\mathbf{p_s}, p_h, p_\eta)$, we see that the *forces* $\dot{\mathbf{p}}_{\mathbf{s}_i}$, $\dot{p}_h$ and $\dot{p}_\eta$ given in (21)-(23) are now velocity-dependent. Because the Störmer-Verlet schemes as well as Beeman's third-order time-integration method are based on the assumption that the forces depend on the coordinates only, the appearance of velocities or momenta on the right-hand side of equations (21)-(23) does not allow one to employ these algorithms directly. The velocity-Störmer-Verlet algorithm can be modified to treat velocity-dependent forces with the help of the Newton-Raphson iteration [65, 89]. The third-order Beeman scheme has to be generalized to a so-called predictor-corrector scheme [90, 91]. The variant due to Refson [91] reads as follows:

(a) $$\mathbf{q}^{n+1} := \mathbf{q}^n + \Delta t m^{-1}\mathbf{p}^n + \frac{\Delta t^2}{6} m^{-1}\left(4\dot{\mathbf{p}}^n - \dot{\mathbf{p}}^{n-1}\right) \tag{56}$$

(b) $$\mathbf{p}_{(p)}^{n+1} := \mathbf{p}^n + \frac{\Delta t}{2}\left(3\dot{\mathbf{p}}^n - \dot{\mathbf{p}}^{n-1}\right) \tag{57}$$

(c) $$\dot{\mathbf{p}}^{n+1} := \mathbf{F}(\mathbf{q}^{n+1}, \mathbf{p}_{(p)}^{n+1}) \tag{58}$$

(d) $$\mathbf{p}_{(c)}^{n+1} := \mathbf{p}^n + \frac{\Delta t}{6}\left(2\dot{\mathbf{p}}^{n+1} + 5\dot{\mathbf{p}}^n - \dot{\mathbf{p}}^{n-1}\right) \tag{59}$$

(e) Replace $\mathbf{p}_{(p)}^{n+1}$ with $\mathbf{p}_{(c)}^{n+1}$. If convergence goto f) else goto (c). (60)

(f) Replace $\mathbf{p}^{n+1}$ with $\mathbf{p}_{(c)}^{n+1}$. For next timestep set $n := n + 1$ and goto (a). (61)

Here, $\mathbf{p}_{(p)}$ and $\mathbf{p}_{(c)}$ represent the *predicted* and *corrected* momenta. Note that in the case of non-Hamiltonian dynamics, the force term $\mathbf{F}(\mathbf{q}^{n+1}, \mathbf{p}_{(p)}^{n+1})$ cannot be derived from the conserved energy function $\mathcal{H}_{\mathrm{NPT}}$ given in equation (24) by $-\nabla_{\mathbf{q}}\mathcal{H}(\mathbf{q}^{n+1}, \mathbf{p}_{(p)}^{n+1})$. The force expression $\mathbf{F}(\mathbf{q}^{n+1}, \mathbf{p}_{(p)}^{n+1})$ is instead given by the right-hand side of the equations of motion (21)-(23). The predictor for the coordinates in step (a) is equivalent to equation (54) and the corrector for the momenta is the same as in equation (55). The expression for the predicted momenta $\mathbf{p}_{(p)}$ in step (b) can also be derived with the help of Taylor expansions. The predictor-corrector cycle (steps (c) to (e)) is iterated until the predicted and corrected momenta $\mathbf{p}_{(p)}$ and $\mathbf{p}_{(c)}$ have converged with respect to a relative precision. This iteration usually needs only two or three cycles in

practice. Furthermore, the expensive evaluation of the coordinate-dependent terms of the force term $\mathbf{F}(\mathbf{q}^{n+1}, \mathbf{p}_{(p)}^{n+1})$ in step (c) has to be performed just once per time step. Only the relatively cheap evaluation of the velocity-dependent terms of the force term has to be computed in every cycle. The iterative time-integration scheme given in equations (56)-(61) now allows to compute the equations of motion (20)-(23) related to an (N, P, T) ensemble.

There are several other time-integration schemes to compute the equations of motions related to Hamiltonian [83] and to non-Hamiltonian [70] dynamics [62, 65, 92]. The accuracy can be improved if either a smaller time step Δt is used, or if a scheme of higher order is employed. The global error of the trajectory $\|\mathbf{q}^n - \mathbf{q}(t_n)\|$ and the momenta $\|\mathbf{p}^n - \mathbf{p}(t_n)\|$ can also be controlled by the choice of the timestep and the order. But note that the global error increases *exponentially* with time, that is, $\|\mathbf{q}^n - \mathbf{q}(t_n)\| \leq C\Delta t^p \frac{e^{M(t_{\text{end}}-t_0)}-1}{M}$ with constants C and M. Therefore, any trajectory computed over long times is meaningless. Nevertheless, long-time molecular dynamics simulations still make sense, because several statistical averages are reproduced very well. For a further discussion on this subject, see [92] or [93]. Beside Beeman's third-order scheme, there are other predictor-corrector and multi-step methods of higher order [90, 88, 94, 95]. Furthermore, there exist partitioned Runge-Kutta algorithms [96, 97]. A general construction method for time-integration schemes is given by operator-splitting [98, 99, 100, 93, 101]. In the case of potentials with different lengthscales and thus timescales, multiple time-step methods like the impulse- or r-Respa schemes can be used to speed up the time-integration [102, 103, 104, 105, 106, 107, 108]. In the case of high-frequency bond length and bond angle potentials, constraints can be applied with the help of the *Shake* [109] or the *Rattle* [110] algorithms. This way, larger time-step sizes can be employed without loosing stability.

2.1.4 Local Minimization

Local minimization of the potential energy with respect to the coordinates is a helpful tool to generate appropriate start configurations for molecular dynamic simulations of polymer-carbon nanotube composites. The problem can be stated as follows: Find a local solution $\mathbf{q}^*$ of the problem:

$$\text{minimize} \quad f(\mathbf{q}), \qquad \text{with } \mathbf{q} \in \mathbb{R}^n. \tag{62}$$

With respect to the (N, P, T) ensemble, we define the function $f : \mathbb{R}^n \to \mathbb{R}$ as the energy function $\mathcal{H}_{\text{NPT}}$ given in equation (24) at a temperature of 0 K

$$f(\mathbf{s}_1, \ldots, \mathbf{s}_N, h) := U(h\mathbf{s}, h) + P_{\text{ext}} \det h\,.$$

Thus, we are concerned with an unconstrained minimization problem in $n = 3N + n_h$ dimensions, where n_h denotes the number of degrees of freedom taken into account from the transformation matrix h. Note that usually only symmetric matrices with $n_h = 6$ are considered, because the absolute orientation of the simulation cell is physically meaningless.

There are several methods for the numerical solution of problem (62) [111, 112, 113, 114]. The so-called *gradient methods* are based on the evaluation of $f(\mathbf{q})$ and

$\nabla_{\mathbf{q}} f(\mathbf{q})$. The so-called *second derivative methods* are additionally based on the Hessian matrix. There are also *derivative-free methods*, which are based on the computation of $f(\mathbf{q})$ only. Because we compute $f(\mathbf{q})$ and $\nabla_{\mathbf{q}} f(\mathbf{q})$ in the framework of a molecular dynamics simulation code anyway, we focus on the gradient based methods in the following.

Iterative *line search* methods are of the form

$$\mathbf{q}^{(k)} := \begin{cases} \mathbf{q}^{(0)} & \text{if } k = 0\,, \\ \mathbf{q}^{(k-1)} + \alpha^{(k-1)} \mathbf{d}^{(k-1)} & \text{if } k > 0\,, \end{cases}$$

where $\mathbf{q}^{(0)}$ is an arbitrary start coordinate and $\alpha^{(k)}$ is a positive step size that minimizes $f(\mathbf{q}^{(k)} + \alpha^{(k)} \mathbf{d}^{(k)})$. Here, finding $\alpha^{(k)}$ is a one-dimensional subproblem. The search directions are usually defined by

$$\mathbf{d}^{(k)} := \begin{cases} -\mathbf{g}^{(0)} & \text{if } k = 0\,, \\ -\mathbf{g}^{(k)} + \beta^{(k)} \mathbf{d}^{(k-1)} & \text{if } k > 0\,, \end{cases}$$

with $\mathbf{g}^{(k)} := \nabla_{\mathbf{q}} f(\mathbf{q}^{(k)})$. In the case of the *steepest descent* method, the scalar $\beta^{(k)}$ is defined as zero for all k. In the case of the nonlinear conjugate gradient method, well-known formula for $\beta^{(k)}$ are due to Fletcher-Reeves [115] and Polak-Ribiére [116]. They are given by

$$\beta_{\mathrm{FR}}^{(k)} := \frac{\mathbf{g}^{(k)^T} \mathbf{g}^{(k)}}{\mathbf{g}^{(k-1)^T} \mathbf{g}^{(k-1)}}\,,$$
$$\beta_{\mathrm{PR}}^{(k)} := \frac{\mathbf{g}^{(k)^T} \left(\mathbf{g}^{(k)} - \mathbf{g}^{(k-1)}\right)}{\mathbf{g}^{(k-1)^T} \mathbf{g}^{(k-1)}}\,.$$

For a strictly convex quadratic function f, these methods are equivalent to the well-known linear *conjugate gradient* scheme due to Hestenes and Stiefel [117].

The steepest descent method tends to converge slowly near a local minimum due to oscillation of the search directions. One the other hand, the nonlinear conjugate gradient schemes assume that the initial conformation is sufficiently close to a local minimum where the potential energy surface is nearly quadratic. Therefore, these methods can be unstable if the start conformation is far away from such a local minimizer. To obtain a robust and efficient unconstrained minimization procedure, the steepest descent method can be used for the first few iterations, and then a conjugate gradient method is invoked. Also, a conjugate gradient scheme with regular restarts can be applied [118, 119]. Here, the search direction $\mathbf{d}^{(k)}$ are periodically or automatically reset to the steepest descent direction $\mathbf{s}^{(k)}$. A Polak-Ribiére based method with automatic restart is performed, if the modified formula

$$\tilde{\beta}_{\mathrm{PR}}^{(k)} := \max\left\{0, \beta_{\mathrm{PR}}^{(k)}\right\} \tag{63}$$

is used [120].

To establish global convergence of line search methods for unconstrained minimization the step size $\alpha^{(k)}$ of the approximate one-dimensional minimization problem has usually to satisfy some appropriate conditions [121, 122]. First there is the so-called *sufficient decrease* or *Armijo* condition

$$f\left(\mathbf{q}^{(k)} + \alpha^{(k)} \mathbf{d}^{(k)}\right) \leq f\left(\mathbf{q}^{(k)}\right) + \mu \alpha^{(k)} \mathbf{g}^{(k)^T} \mathbf{d}^{(k)} \tag{64}$$

to avoid too long step sizes [123]. Second there is the so-called *curvature* condition

$$\left(\nabla_{\mathbf{q}} f\left(\mathbf{q}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)}\right)\right)^T \mathbf{d}^{(k)} \geq \nu \mathbf{g}^{(k)^T}\mathbf{d}^{(k)} \tag{65}$$

to avoid too short step sizes. Here, the two parameters μ, ν with $0 < \mu < \nu < 1$ are user-defined and can be employed to weight conditions (64) and (65) properly. The Armijo condition (64) together with the curvature condition (65) are also called *Wolfe* conditions [124, 125]. Moreover, when the curvature condition (65) is replaced by

$$\left|\left(\nabla_{\mathbf{q}} f\left(\mathbf{q}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)}\right)\right)^T \mathbf{d}^{(k)}\right| \leq \nu \left|\mathbf{g}^{(k)^T}\mathbf{d}^{(k)}\right| , \tag{66}$$

we obtain the so-called *strong Wolfe* conditions. Like in the Wolfe conditions, the Armijo condition fixes an upper limit on acceptable new function values, and the second condition (66) imposes a lower bound on the step size $\alpha^{(k)}$. Note that the strong Wolfe conditions lead to an exact line search for ν chosen arbitrarily close to zero.

Now, a backtracking line search algorithm to determine a step size $\alpha^{(k)}$ that fulfills the Armijo condition (64) can be obtained as follows: After an initial value $\alpha_0^{(k)}$ is chosen, the step size $\alpha_i^{(k)}$ is iteratively replaced by a new value $\alpha_{i+1}^{(k)}$ from the interval $\left[\gamma_1\alpha_i^{(k)}, \gamma_2\alpha_i^{(k)}\right]$, with $0 < \gamma_1 < \gamma_2 < 1$, until the Armijo condition (64) is satisfied. Here, a simple strategy for the choice of $\alpha_{i+1}^{(k)}$ is given by $\alpha_{i+1}^{(k)} := \delta\alpha_i^{(k)}$ with $0 < \delta < 1$. Because we have to determine the values $f(\mathbf{q}^{(k)} + \alpha_i^{(k)}\mathbf{d}^{(k)})$ and the slope $\mathbf{g}^{(k)^T}\mathbf{d}^{(k)}$ to test the Armijo condition (64) and we have to compute the values $f(\mathbf{q}^{(k)})$ anyway, we can exploit these dates to obtain an improved strategy: We interpolate the function f by a quadratic polynomial p and obtain the new step size $\alpha_{i+1}^{(k)}$ as a minimizer of that polynomial p. Moreover, we can use an additional previous function value $f(\mathbf{q}^{(k)} + \alpha_{i-1}^{(k)}\mathbf{d}^{(k)})$ or the slope $\left(\nabla_{\mathbf{q}} f\left(\mathbf{q}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)}\right)\right)^T \mathbf{d}^{(k)}$ to interpolate the function f by a cubic polynomial p and can determine a new value $\alpha_{i+1}^{(k)}$ as a minimizer of that p. If the minimizer of the respective interpolation polynomial p is not in the interval $\left[\gamma_1\alpha_i^{(k)}, \gamma_2\alpha_i^{(k)}\right]$, then simply one of the boundaries of the interval can be chosen as $\alpha_{i+1}^{(k)}$. Note that we also have to compute the slope $\left(\nabla_{\mathbf{q}} f\left(\mathbf{q}^{(k)} + \alpha^{(k)}\mathbf{d}^{(k)}\right)\right)^T \mathbf{d}^{(k)}$ to verify the curvature condition (65) or the stronger curvature condition (66). Here, a backtracking algorithm to find a step size $\alpha^{(k)}$, which also fulfills theses curvature conditions, can be performed in an analogous way, but the details are more complex. For a further discussion see [111, 112, 113, 114].

Note that for conjugate gradient methods with exact line search the order of convergence is linear and cannot be superlinear [126]. There exist other algorithms with a local convergence order better than linear that still need a linear order storage only, like limited memory preconditioned conjugate gradient methods, *quasi-Newton-Raphson* schemes like the *DFP* [127, 128], or the *BFGS* methods [129, 130, 131, 132, 133]. However, these techniques are more complex and their storage requirements are relatively demanding, because they are based on an approximate inverse of the Hessian matrix. Note that, in the framework of our simulations, we have to deal with hundreds of thousands of atoms to generate reference systems of polymer-carbon

nanotube composites. Because the modified Polak-Ribiére method given in equation (63) compromises well between convergence and storage requirements, we use it in the following. Here, we perform a backtracking algorithm together with cubic interpolation to fulfill the strong Wolfe condition for every step size $\alpha^{(k)}$.

2.2 Potentials for Hydrocarbons

In the framework of first principles methods, like the Born-Oppenheimer molecular dynamics and the Car-Parrinello molecular dynamics [134], the nuclei move on a potential energy surface given by the electronic Schrödinger equation [135]. Due to the *curse of dimension* of the electronic Schrödinger equation for many-body systems, these approaches are only useful for a small number of particles. To consider large systems, as in the case of polymer-carbon nanotube composites, only classical molecular dynamics can be performed. Here, the global potential energy surface is approximated by a truncated expansion of many-body contributions

$$U^{exact} \approx U(\mathbf{r}_1, \ldots, \mathbf{r}_N) = \sum_{i=1}^{N} u_1(\mathbf{r}_i) + \sum_{i<j}^{N} u_2(\mathbf{r}_i, \mathbf{r}_j) + \sum_{i<j<k}^{N} u_3(\mathbf{r}_i, \mathbf{r}_j, \mathbf{r}_k) + \cdots .$$

This way, the number of degrees of freedom is reduced to the order of the number of atoms. The interaction potentials u_n usually are simple analytical expressions, which are parametrized by empirical data due to experiments or first principles calculations. In the following, we discuss two different models to represent a carbon nanotube together with a polyethylene matrix. In the atomistic model, which we denote here as *model I*, we represent the nanotube as well as the polyethylene matrix by a *reactive empirical bond order* (REBO) potential due to Brenner [59]. In the united-atom approach, which we denote as *model II*, we represent the carbon nanotube with Brenner's potential too, but the polyethylene matrix is represented by a so-called *expanded collapsed atomic model* with static bond relations [61].

2.2.1 Reactive Atomistic Model (Model I)

Here, we use a so-called atomistic model in which all the atoms of the nanototube as well as all the atoms of the monomeric units of the polymer are treated explicitly. To model the short-range chemical interactions, we employ Brenner's potential. The design of this REBO potential for hydrocarbons is based on a bonding formalism due to Abell and Tersoff [136, 137, 138, 139, 140], which relies on Pauling's bond order ideas [141, 142]. Abell's and Tersoff's potential allows the formation and the breaking of bonds. Furthermore, carbon-carbon single, double, and triple bond-lengths and energies are well described. But in the special situation, where nonlocal effects are involved, this potential leads to unphysical behavior, because only local near-neighbor interactions are taken into account. As an example, consider the case of a carbon atom that possesses three nearest neighbors and that is bonded to an other carbon atom which has four nearest neighbors. Then the evaluation of Abell's and Tersoff's potential would result in something between a single and a double bond [59]. But such a configuration would be better modeled by a single bond plus a radical orbital. By taking nonlocal effects into account, Brenner's approach [59] is able to accurately describe covalent bonding, including radicals, and can distinguish

conjugated and nonconjugated double bonds. In fact, Brenner's REBO potential model has been successfully applied to simulate indentation and friction at nanoscale [143, 144, 145, 146, 147, 148, 149], hydrogen abstraction reactions of organic films [150], adhesion and compression of diamond surfaces [151], as well as abstraction of hydrogen from a diamond surface [152]. Moreover, several reactions with surfaces have been investigated [153, 154, 155], like the reaction of a surface molecule with a gas-phase molecule [156]. Furthermore, the REBO potential model of Brenner has been widely used to study the formation of carbon fullerenes and their properties [157, 158, 19, 159], the energetics of carbon nanotubes [19], the growth of carbon nanotubes [160, 161, 162, 163, 164, 165], carbon nanotube fracture [31], and the mechanics of carbon nanotubes [26, 28, 21, 29, 33, 34] and filled carbon nanotubes [166, 167, 30, 32]. In particular, Brenner's potential has been successfully applied to examine carbon nanotube composites [168], like carbon nanotube reinforced polymer [39, 169] and functionalized carbon nanotube reinforced polymer [37, 38, 170]. To also characterize nonbonded van der Waals interactions, we additionally apply a Lennard-Jones potential parametrized for hydrocarbons [171, 168]. Then, the resulting REBO potential model is given as a sum over bonds

$$U_{\mathrm{I}} = \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j>i}}^{N} \underbrace{f_{ij}(r_{ij}) \left(u_R(r_{ij}) - \bar{B}_{ij} u_A(r_{ij}) \right)}_{u_B(r_{ij})} + \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j>i}}^{N} u_W(r_{ij}) \,. \tag{67}$$

Here, Brenner's bond potential term u_B consists of a pair potential term u_R to model interatomic core-core repulsive interactions and a pair potential term u_A to describe the attractive interactions between valence electrons and cores. The factor $\bar{B}_{ij}$ is a so-called empirical bond-order term, which modulates valence electron densities and depends on the bond lengths and the angles. Due to the fact that the factor $\bar{B}_{ij}$ varies subjected to the local configuration of atoms, Brenner's potential is a reactive many-body potential model. The potential term u_B is restricted to nearest neighbor interactions by a smooth truncation function f_{ij}. The term u_W denotes the contribution from van der Waals energy modeled by the Lennard-Jones potential. It has a support different from that of the short-range diatomic potential term u_B.

To be precise, the repulsive and the attractive pair terms u_R and u_A are defined as

$$\begin{aligned} u_R(r_{ij}) &= \frac{D_{ij}^{(e)}}{S_{ij}-1} e^{-\sqrt{2S_{ij}}\beta_{ij}\left(r_{ij}-R_{ij}^{(e)}\right)} \,, \\ u_A(r_{ij}) &= \frac{D_{ij}^{(e)}}{S_{ij}-1} S_{ij} e^{-\sqrt{\frac{2}{S_{ij}}}\beta_{ij}\left(r_{ij}-R_{ij}^{(e)}\right)} \,, \end{aligned}$$

with parameters $D_{ij}^{(e)}$, $R_{ij}^{(e)}$ and β_{ij} S_{ij}. These constant parameters are specified by the type of the atoms i and j (i.e. they are dependent on whether i and j are carbon or hydrogen atoms). The possible combinations can be expressed by $D_{ij}^{(e)} \in \left\{ D_{\mathrm{CC}}^{(e)}, D_{\mathrm{CH}}^{(e)}, D_{\mathrm{HC}}^{(e)}, D_{\mathrm{HH}}^{(e)} \right\}$ and analogously for $R_{ij}^{(e)}$, β_{ij} and S_{ij}. In particular, the values of these constants are pairwise symmetric concerning the combination of atom types. The detailed values for hydrocarbons are given in Table 1. Note that in the case of $B_{ij} \equiv 1$ and $f_{ij} \equiv 1$ the potential term u_B in equation (67) reduces to

Table 1: Pair potential parameters for the Brenner potential [59].

Carbon			Hydrogen			Hydrocarbons		
$D^{(e)}_{\mathrm{CC}}$	=	6.325 eV	$D^{(e)}_{\mathrm{HH}}$	=	4.7509 eV	$D^{(e)}_{\mathrm{CH}}$=$D^{(e)}_{\mathrm{HC}}$	=	3.6422 eV
$R^{(e)}_{\mathrm{CC}}$	=	1.315 Å	$R^{(e)}_{\mathrm{HH}}$	=	0.74144 Å	$R^{(e)}_{\mathrm{CH}}$=$R^{(e)}_{\mathrm{HC}}$	=	1.1199 Å
β_{CC}	=	1.5 Å^{-1}	β_{HH}	=	1.9436 Å^{-1}	β_{CH}=β_{HC}	=	1.9583 Å^{-1}
S_{CC}	=	1.29	S_{HH}	=	2.3432	S_{CH}=S_{HC}	=	1.7386
$R^{(1)}_{\mathrm{CC}}$	=	1.7 Å	$R^{(1)}_{\mathrm{HH}}$	=	1.1 Å	$R^{(1)}_{\mathrm{CH}}$=$R^{(1)}_{\mathrm{HC}}$	=	1.3 Å
$R^{(2)}_{\mathrm{CC}}$	=	2.0 Å	$R^{(2)}_{\mathrm{HH}}$	=	1.7 Å	$R^{(2)}_{\mathrm{CH}}$=$R^{(2)}_{\mathrm{HC}}$	=	1.8 Å

the potential function

$$u_{\mathrm{Morse}}(r_{ij}) = \frac{D^{(e)}_{ij}}{S_{ij}-1} e^{-\sqrt{2S_{ij}}\beta_{ij}\left(r_{ij}-R^{(e)}_{ij}\right)} - \frac{D^{(e)}_{ij}}{S_{ij}-1} S_{ij} e^{-\sqrt{\frac{2}{S_{ij}}}\beta_{ij}\left(r_{ij}-R^{(e)}_{ij}\right)}, \qquad (68)$$

which corresponds for the case $S_{ij} = 2$ to the usual diatomic Morse potential model [172]. Independent of the fitting parameters $S_{ij} > 0$ and $\beta_{ij} > 0$, the energy function (68) has a minimum of $-D^{(e)}_{ij}$ at the equilibrium distance $R^{(e)}_{ij}$ and the limit for $r_{ij} \to \infty$ is zero. Note that in contrast to the Coulomb potential, the Morse potential is finite at $r_{ij} = 0$. But for typical values of the parameters D_{ij}, S_{ij}, and β_{ij}, the Morse potential takes a value of more than $100D_{ij}$ at $r_{ij} = 0$. Thus, its effect for small distances $r_{ij} \approx 0$ is nearly equivalent to that of the Coulomb case. Furthermore, Brenner's pair potential term u_B is restricted to immediate neighbors or bonded atoms by a factor f_{ij}. This smoothed truncation function f_{ij} reads as

$$f_{ij}(r_{ij}) = \begin{cases} 1 & \text{if } r_{ij} \le R^{(1)}_{ij}, \\ \frac{1}{2}\left[1 + \cos\left(\pi \frac{r_{ij}-R^{(1)}_{ij}}{R^{(2)}_{ij}-R^{(1)}_{ij}}\right)\right] & \text{if } R^{(1)}_{ij} < r_{ij} < R^{(2)}_{ij}, \\ 0 & \text{if } R^{(2)}_{ij} \le r_{ij}, \end{cases} \qquad (69)$$

with cutoff parameters $R^{(1)}_{ij}$, and $R^{(2)}_{ij}$. Like the parameters $D^{(e)}_{ij}$, $R^{(e)}_{\mathrm{CC}}$, β_{CC} and S_{CC}, the cutoff parameters $R^{(1)}_{ij}$ and $R^{(2)}_{ij}$ are only dependent on the type of the atoms i and j; see Table 1.

The most interesting term of Brenner's model is the many-body empirical bond-order term $\bar{B}_{ij}$. In analogy to the approach of Abell and Tersoff, it is defined as the average of bond-order terms subjected to each atom of a bond. In Brenner's model nonlocal effects are included by using additional empirical correction terms. These correction terms are added to avoid the overbinding of radicals as well as to deal with conjugated versus nonconjugated bonds. To be precise, to describe the bonding configuration of a carbon atom i, the number $N^{(\mathrm{C})}_i$ of carbon atoms and the number $N^{(\mathrm{H})}_i$ of hydrogen atoms bonded to a carbon atom i are approximated by

$$N^{(\mathrm{C})}_i = \sum_{j \in C} f_{ij}(r_{ij}) \qquad \text{and} \qquad N^{(\mathrm{H})}_i = \sum_{j \in H} f_{ij}(r_{ij}) \qquad \text{for } i \in C. \qquad (70)$$

Here, C denotes the set of indices of carbon atoms and H denotes the set of indices of hydrogen atoms, respectively. Note that the smoothed truncation function f_{ij} can also be seen as an enumerator for bonds: Inside the sphere defined by the radius $R_{ij}^{(1)}$ around particle i the function is just one and outside the sphere defined by the radius $R_{ij}^{(2)}$ it vanishes identically. In addition, we define the total number of atoms, which interact with a carbon atom i, in the form

$$N_i^{(t)} = N_i^{(\mathrm{C})} + N_i^{(\mathrm{H})} \qquad \text{for } i \in C.$$

Now, we focus on conjugated bonds. For example, if all neigbors k of a carbon-carbon bond are carbon bonds and have a coordination less than 4 (i.e. $N_k^{(t)} < 4$), then this carbon-carbon bond is considered as being part of a conjugated system [59]. Hence, we define a further smooth enumeration function

$$N_{ij}^{(c)} = 1 + \sum_{\substack{k \in C \\ k \neq i \wedge k \neq j}} f_{ik}(r_{ik}) F\left(N_k^{(t)} - f_{ik}(r_{ik})\right) + \sum_{\substack{l \in C \\ l \neq i \wedge l \neq j}} f_{jl}(r_{jl}) F\left(N_l^{(t)} - f_{jl}(r_{jl})\right) \tag{71}$$

for each carbon-carbon bond ($i \in C, j \in C$), where

$$F(z) = \begin{cases} 1 & \text{if } z \leq 2, \\ \frac{1}{2}\left[1 + \cos\left(\pi(z-2)\right)\right] & \text{if } 2 < z < 3, \\ 0 & \text{if } 3 \leq z, \end{cases}$$

is a smooth truncation function similar to f_{ij} given in equation (69). This way, a bond (i, j) is not part of a conjugated system, if $N_{ij}^{(c)} = 1$. For $N_{ij}^{(c)} \geq 2$, it is detected as part of a conjugated system. Now, the enumeration functions $N_i^{(\mathrm{C})}$, $N_i^{(\mathrm{H})}$, $N_i^{(t)}$ and $N_{ij}^{(c)}$ can be exploited to include second-neighbor effects and thus to adjust the potential to various known bonding configurations by correction functions. To this end, we define a bond-order term B_{ij}, which is subjected to the atom i of a bond (i, j), in the following way:

$$B_{ij} = \left(1 + \sum_{\substack{k=1 \\ k \neq i \wedge k \neq j}}^{N} G_i(\theta_{ijk}) f_{ik}(r_{ik}) e^{\alpha_{ijk}\left(r_{ij} - R_{ij}^{(e)} - r_{ik} + R_{ik}^{(e)}\right)} + H_{ij}\left(N_i^{(\mathrm{C})}, N_i^{(\mathrm{H})}\right) \right)^{-\delta_i}, \tag{72}$$

where H_{ij} is a correction term, which in particular smooths the transit from a bonded to a nonbonded state. It is defined as a two-dimensional cubic spline that depends on the atom type and the coordination numbers of atom i and atom j, respectively; see Table 2. The parameter δ_i depends on the type of atom i and the parameter α_{ijk} depends on the types of atoms i, j and k, respectively; see Table 3. Furthermore G_i is a function of the type of atom i and the angle θ_{ijk} between the bond distance vectors $\mathbf{r}_{ij}$ and $\mathbf{r}_{ik}$; see Table 3. Note that the bond-order terms B_{ij} and B_{ji} subjected to each atom of a bond (i, j) are not symmetric in i and j. Now, the final symmetric bond-oder term $\bar{B}_{ij}$ is defined in the form

$$\bar{B}_{ij} = \frac{1}{2}\left(B_{ij} + B_{ji}\right) + K\left(N_i^{(t)}, N_j^{(t)}, N_{ij}^{(c)}\right), \tag{73}$$

Table 2: Function values and values of the partial derivatives for the two dimensional spline $H_{ij}\left(N_i^{(\mathrm{H})}, N_i^{(\mathrm{C})}\right)$ in equation (72) of the Brenner potential model [59, 60]. The parameters not given are equal to zero.

$\left(N_i^{(\mathrm{H})}, N_i^{(\mathrm{C})}\right)$	H_{CC}	H_{CH}	$\frac{\partial H_{\mathrm{CH}}}{\partial N_i^{(\mathrm{H})}}$	$\frac{\partial H_{\mathrm{CH}}}{\partial N_i^{(\mathrm{C})}}$
$(1,0)$		-0.0760		
$(2,0)$	-0.0070	-0.2163	-0.13075	-0.07655
$(3,0)$	0.0119	-0.3375		
$(0,1)$		-0.1792		
$(0,2)$		-0.2407	-0.13075	-0.07655
$(0,3)$		-0.3323		
$(1,1)$	-0.0175	-0.2477	-0.07640	-0.12805
$(1,2)$	0.0115	-0.3321		
$(2,1)$	0.0118	-0.3320		

Table 3: The parameters for the bond-order term B_{ij} given in equation (72) of the Brenner potential model [59] and the parameters for the function $G_{\mathrm{C}}(\theta) = a_0\left(1 + c_0^2/d_0^2 - c_0^2/\left(d_0^2+(1+\cos\theta)^2\right)\right)$. Note that the function G_{H} is constant.

Carbon			Hydrogen			Hydrocarbons		
δ_{C}	=	0.80469	δ_{H}	=	0.80469			
α_{CCC}	=	$0.0\,\text{Å}^{-1}$	α_{HHH}	=	$3.0\,\text{Å}^{-1}$	$\alpha_{\mathrm{HHC}}=\alpha_{\mathrm{CHH}}$	=	$3.0\,\text{Å}^{-1}$
						$\alpha_{\mathrm{HCH}}=\alpha_{\mathrm{HCC}}$	=	$3.0\,\text{Å}^{-1}$
						$\alpha_{\mathrm{CHC}}=\alpha_{\mathrm{CCH}}$	=	$0.0\,\text{Å}^{-1}$
a_0	=	0.011304	G_{H}	=	4.0			
c_0	=	19						
d_0	=	2.5						

Table 4: Function values and values of the partial derivatives for the three-dimensional spline $K(n_1, n_2, n_3)$ in equation (73) of the Brenner potential model [59, 60]. The three dimensional spline has to fulfill the symmetry conditions $K\left(N_i^{(t)}, N_j^{(t)}, N_{ij}^{(c)}\right) = K\left(N_j^{(t)}, N_i^{(t)}, N_{ij}^{(c)}\right)$ and $\frac{\partial K}{\partial n_1}\left(N_i^{(t)}, N_j^{(t)}, N_{ij}^{(c)}\right) = \frac{\partial K}{\partial n_2}\left(N_j^{(t)}, N_i^{(t)}, N_{ji}^{(c)}\right)$. Furthermore, all function values $K\left(N_i^{(t)}, N_j^{(t)}, N_{ij}^{(c)}\right)$ with $N_{ij}^{(c)} > 2$ are equal to $K\left(N_i^{(t)}, N_j^{(t)}, 2\right)$. All other parameters that are not given are equal to zero.

$\left(N_i^{(t)}, N_j^{(t)}, N_{ij}^{(c)}\right)$	K	$\left(N_i^{(t)}, N_j^{(t)}, N_{ij}^{(c)}\right)$	$\frac{\partial K}{\partial n_1}$
(2, 3, 1)	-0.0465	(2, 3, 1)	0.05650
(2, 3, 2)	-0.0465	(2, 3, 2)	0.05650
(1, 2, 2)	-0.0355	(1, 2, 2)	0.02225
(1, 1, 1)	0.1511	(3, 1, 1)	-0.11600
(2, 2, 1)	0.0750	(3, 2, 1)	-0.13205
(1, 2, 1)	0.0126	(3, 1, 2)	-0.06100
(1, 3, 1)	-0.1130	(2, 4, 2)	-0.03775
(1, 3, 2)	-0.1130	(3, 4, 2)	0.05650
(0, 3, 1)	-0.1220	(3, 4, 1)	0.05650
(0, 3, 2)	-0.1220	(3, 2, 2)	-0.10650
(0, 2, 2)	-0.0445	(2, 0, 1)	-0.11600
(0, 2, 1)	0.0320	(2, 1, 1)	-0.13205
(0, 1, 1)	0.1100	(2, 0, 2)	-0.06100
(1, 1, 2)	0.0074	(1, 3, 2)	0.03775
		(2, 1, 2)	-0.6020

where K is a three-dimensional cubic spline, which is in particular symmetric in i and j. Its data are given in Table 4. The correction term K depends on the bonding connectivity and thus allows to fix the overbinding of radicals as well as to handle conjugated versus nonconjugated bonds.

Now we consider the potential function u_W. To obtain here a continuously differentiable potential u_W we use cubic spline functions $S_{ij}^{(1)}$ and $S_{ij}^{(2)}$ and define

$$u_W(r_{ij}) = \begin{cases} 0 & \text{if } r_{ij} \le R_{ij}^{(2)}, \\ S^{(1)}(r_{ij}) & \text{if } R_{ij}^{(2)} < r_{ij} \le R_{ij}^{(3)} \\ u_{\mathrm{LJ}}(r_{ij}) & \text{if } R_{ij}^{(3)} < r_{ij} \le R_{ij}^{(4)} \\ S^{(2)}(r_{ij}) & \text{if } R_{ij}^{(4)} < r_{ij} < R_{ij}^{(5)}, \\ 0 & \text{if } R_{ij}^{(5)} \le r_{ij}, \end{cases} \tag{74}$$

where u_{LJ} denotes the Lennard-Jones potential

$$u_{\mathrm{LJ}}(r_{ij}) = 4\epsilon_{ij}\left(\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right).$$

The Lennard-Jones parameters ϵ_{ij} and σ_{ij} and the radii $R_{ij}^{(2)}$, $R_{ij}^{(3)}$, $R_{ij}^{(4)}$, and $R_{ij}^{(5)}$ are

Table 5: The parameters for the Lennard-Jones/van der Waals interaction u_w. The values for ϵ_{ij}, σ_{ij}, $R_{ij}^{(2)}$, $R_{ij}^{(3)}$ and $R_{ij}^{(5)}$ are taken from [171, 166]. Here the cutoff parameter $R_{ij}^{(2)}$ corresponds to the outer cutoff radius of the Brenner potential given in Table 1. The parameters ϵ_{CH} and σ_{CH} are obtained by the Lorentz-Berthelot mixing rules (75).

Carbon			Hydrogen		
ϵ_{CC}	=	0.0042038 eV	ϵ_{HH}	=	0.0058901 eV
σ_{CC}	=	3.37 Å	σ_{HH}	=	2.91 Å
$R_{\mathrm{CC}}^{(2)}$	=	2.0 Å	$R_{\mathrm{HH}}^{(2)}$	=	1.7 Å
$R_{\mathrm{CC}}^{(3)}$	=	3.20 Å	$R_{\mathrm{HH}}^{(3)}$	=	2.76 Å
$R_{\mathrm{CC}}^{(4)}$	=	9.875 Å	$R_{\mathrm{HH}}^{(4)}$	=	9.875 Å
$R_{\mathrm{CC}}^{(5)}$	=	10.0 Å	$R_{\mathrm{HH}}^{(5)}$	=	10.0 Å

Hydrocarbons		
$\epsilon_{\mathrm{CH}}=\epsilon_{\mathrm{HC}}$	=	0.0049760 eV
$\sigma_{\mathrm{CH}}=\sigma_{\mathrm{HC}}$	=	3.14 Å
$R_{\mathrm{CH}}^{(2)}=R_{\mathrm{HC}}^{(2)}$	=	1.8 Å
$R_{\mathrm{CH}}^{(3)}=R_{\mathrm{HC}}^{(3)}$	=	2.98 Å
$R_{\mathrm{CH}}^{(4)}=R_{\mathrm{HC}}^{(4)}$	=	9.875 Å
$R_{\mathrm{CH}}^{(5)}=R_{\mathrm{HC}}^{(5)}$	=	10.0 Å

given in Table 5. The parameters ϵ_{CH} and σ_{CH} are obtained by the Lorentz-Berthelot mixing rules

$$\epsilon_{ij} = \sqrt{\epsilon_{ii}\epsilon_{jj}}\,, \qquad \sigma_{ij} = (\sigma_{ii} + \sigma_{jj})/2\,. \tag{75}$$

To obtain a continuously differentiable potential u_W, we use in (74) cubic splines $S_{ij}^{(1)}$ and $S_{ij}^{(2)}$ with

$$\begin{aligned}
&S_{ij}^{(1)}\left(R_{ij}^{(2)}\right) = 0\,, && \frac{dS_{ij}^{(1)}}{dr}\left(R_{ij}^{(2)}\right) = 0\,,\\
&S_{ij}^{(1)}\left(R_{ij}^{(3)}\right) = u_{\mathrm{LJ}}\left(R_{ij}^{(3)}\right)\,, && \frac{dS_{ij}^{(1)}}{dr}\left(R_{ij}^{(3)}\right) = \frac{du_{\mathrm{LJ}}}{dr}\left(R_{ij}^{(3)}\right)\,,\\
&S_{ij}^{(2)}\left(R_{ij}^{(4)}\right) = u_{\mathrm{LJ}}\left(R_{ij}^{(4)}\right)\,, && \frac{dS_{ij}^{(2)}}{dr}\left(R_{ij}^{(4)}\right) = \frac{du_{\mathrm{LJ}}}{dr}\left(R_{ij}^{(4)}\right)\,,\\
&S_{ij}^{(2)}\left(R_{ij}^{(5)}\right) = 0\,, && \frac{dS_{ij}^{(2)}}{dr}\left(R_{ij}^{(5)}\right) = 0\,.
\end{aligned}$$

Note that the potential $u_W(r_{ij})$ is non-zero for $R_{ij}^{(2)} \le r_{ij} \le R_{ij}^{(5)}$ only. It thus has a support different from that of u_B in equation (67).

2.2.2 United-Atom Model (Model II)

In our second model, we use a united-atom approach to represent the polyethylene matrix. We employ the so-called expanded collapsed atomic model [61], where the monomeric units $\tilde{\mathrm{C}} := \mathrm{CH}_2$ of a polyethylene chain within the matrix are treated as single spheres. In contrast to the reactive Brenner potential model, the bond relations are static here, and thus no breaking or forming of bonds within the polyethylene matrix are possible. But as we will discuss in the next paragraph, the united-atom approach with static bond relations allows more efficient computations than Brenner's REBO potential.

The model includes the usual bond stretching, bond angle bending, and torsional potentials for the intramolecular interactions within the polyethylene matrix. Altogether, the potential for the the bonded forces within the linear chains of monomeric units $\tilde{\mathrm{C}}$ reads as

$$U_{\mathrm{UA}} = \sum_{(i,j)\in\mathcal{B}} u_{\mathrm{bond}}(r_{ij}) + \sum_{(i,j,k)\in\mathcal{A}} u_{\mathrm{ang}}(\theta_{ijk}) + \sum_{(i,j,k,l)\in\mathcal{T}} u_{\mathrm{tor}}(\phi_{ijkl})\,, \tag{76}$$

where θ_{ijk} denotes the angle between $\mathbf{r}_{ij}$ and $\mathbf{r}_{ik}$ and ϕ_{ijkl} denotes the torsional or dihedral angle $\sphericalangle(\mathbf{r}_{ij} \times \mathbf{r}_{jk}, \mathbf{r}_{jk} \times \mathbf{r}_{kl})$; see also Figure 6. Here, the harmonic bond stretching potential is given in the form

$$u_{\mathrm{bond}}(r) = b_1 \left(r - r_{\mathrm{bond}}\right)^2$$

and the bond angle bending potential is defined by

$$u_{\mathrm{ang}}(\theta) = a_1 \left(\theta - \theta_{\mathrm{ang}}\right)^2 - a_2 \left(\theta - \theta_{\mathrm{ang}}\right)^3 .$$

The equilibrium bond length r_{bond}, the equilibrium bond angle θ_{ang} and the force constants b_1, a_1 and a_2 are given in Table 6. Furthermore, the torsional or dihedral potential [173] reads as

$$u_{\mathrm{tor}}(\phi) = c_1 \left(d_0 + d_1 \cos\phi + d_2 \cos^2\phi + d_3 \cos^3\phi + d_4 \cos^4\phi + d_5 \cos^5\phi\right) ,$$

with parameters given in Table 6.

We now model a polyethylene matrix with the united-atom approach. Here, each CH_2 unit of a polyethylene molecule is treated as a single particle $\tilde{\mathrm{C}}$. Thus, each polyethylene molecule corresponds to a set of particles of $\tilde{\mathrm{C}}$'s that are bonded in linear order and form a chain. We now discuss how to treat N of such linear chains, where the nth chain is composed of N_n monomeric units $\tilde{\mathrm{C}}$. To this end, we define the static bond relations which correspond to the bond stretching forces by the set

$$\mathcal{B} := \big\{(1,2),(2,3),\ldots,(N_1-1,N_1),(N_1+1,N_1+2),\ldots(\sum_{m=1}^{N} N_m - 1, \sum_{m=1}^{N} N_m)\big\}\,,$$

which is just the set of index pairs of consecutively bonded particles. In an analogous way, we define a set of index triplets

$$\begin{aligned}\mathcal{A} := \big\{&(2,1,3),(3,2,4),\ldots,(N_1-1,N_1-2,N_1),\\ &(N_1+2,N_1+1,N_1+3),\ldots(\sum_{m=1}^{N} N_m - 1, \sum_{m=1}^{N} N_m - 2, \sum_{m=1}^{N} N_m)\big\}\end{aligned}$$

Table 6: Parameters for the bond stretching, the bond angle bending, and the torsional or dihedral potentials to model the intramolecular interaction of a linear polyethylene chain in the framework of an united-atom approach [61, 174, 175].

Bond stretching			Bond angle bending		
r_{bond}	=	$1.53\,\text{Å}$	θ_{ang}	=	$109.47\,\text{deg}$
b_1	=	$13.74302\,\text{eV/Å}^2$	a_1	=	$5.2038993 \times 10^{-4}\,\text{eV/deg}^2$
			a_2	=	$4.9955775 \times 10^{-6}\,\text{eV/deg}^3$

Torsional		
c_1	=	$0.010364269\,\text{eV}$
d_0	=	9.279
d_1	=	-12.136
d_2	=	-13.120
d_3	=	3.060
d_4	=	26.240
d_5	=	31.594

and a set of index quadruples

$$\begin{aligned}\mathcal{T} := \big\{&(1,2,3,4),(2,3,4,5),\ldots,(N_1-3,N_1-2,N_1-1,N_1),\\ &(N_1+1,N_1+2,N_1+3,N_1+4),\ldots\\ &\ldots\Big(\sum_{m=1}^{N} N_m - 3, \sum_{m=1}^{N} N_m - 2, \sum_{m=1}^{N} N_m - 1, \sum_{m=1}^{N} N_m\Big)\big\}\,,\end{aligned}$$

which correspond to the bond angle and torsional static relations, respectively. These sets are then used to invoke the bond stretching potential, the bond angle bending potential and the torisional potential in (76).

To model the van der Waals interactions between the monomeric units $\tilde{\text{C}}$, we additionally apply a smoothed Lennard-Jones potential

$$u_{\text{sLJ}}(r_{ij}) = \begin{cases} u_{\text{LJ}}(r_{ij}) & \text{for } r_{ij} \le R_{ij}^{(4)} \\ S^{(2)}(r_{ij}) & \text{for } R_{ij}^{(4)} < r_{ij} < R_{ij}^{(5)} \\ 0 & \text{for } R_{ij}^{(5)} \le r_{ij} \end{cases} \tag{77}$$

between the $\tilde{\text{C}}$'s in different polyethylene molecule chains. Within a polyethylene molecule chain, we apply this potential if the $\tilde{\text{C}}$'s are three or more neighbors apart on the chain. Note that the carbon nanotube is modeled as before using Brenner's potential. To model the interaction between the $\tilde{\text{C}}$'s of the polyethylene and the carbon atoms of the nanotube, we again employ the smoothed Lennard-Jones potential (77) with the parameters given in table 7.

2.2.3 Efficient Implementation

To study the mechanical properties of polymer-carbon nanotube composites, we have to perform molecular dynamics simulations for large systems over long times. To this

Table 7: The parameters for the Lennard-Jones van der Waals interaction between the monomeric units $\tilde{\mathrm{C}}$ of the polyethylene and the carbon nanotube atoms C. The parameters $\epsilon_{\tilde{\mathrm{C}}\mathrm{C}}$ and $\sigma_{\tilde{\mathrm{C}}\mathrm{C}}$ can be derived from the Lorentz-Berthelot mixing rules given in equation (75).

Monomeric units			Matrix - Nanotube		
$\epsilon_{\tilde{\mathrm{C}}\tilde{\mathrm{C}}}$	=	$6.2040512 \times 10^{-3}\,\mathrm{eV}$	$\epsilon_{\tilde{\mathrm{C}}\mathrm{C}} = \epsilon_{\mathrm{C}\tilde{\mathrm{C}}}$	=	$5.5562129 \times 10^{-3}\,\mathrm{eV}$
$\sigma_{\tilde{\mathrm{C}}\tilde{\mathrm{C}}}$	=	$3.9230\,\text{Å}$	$\sigma_{\tilde{\mathrm{C}}\mathrm{C}} = \sigma_{\mathrm{C}\tilde{\mathrm{C}}}$	=	$3.6465\,\text{Å}$
$R^{(4)}_{\tilde{\mathrm{C}}\tilde{\mathrm{C}}}$	=	$9.875\,\text{Å}$	$R^{(4)}_{\tilde{\mathrm{C}}\mathrm{C}} = R^{(4)}_{\mathrm{C}\tilde{\mathrm{C}}}$	=	$9.875\,\text{Å}$
$R^{(5)}_{\tilde{\mathrm{C}}\tilde{\mathrm{C}}}$	=	$10.0\,\text{Å}$	$R^{(5)}_{\tilde{\mathrm{C}}\mathrm{C}} = R^{(5)}_{\mathrm{C}\tilde{\mathrm{C}}}$	=	$10.0\,\text{Å}$

end, we need an efficient implementation of *model I* and *model II*. This involves an efficient implementation of the united-atom potential with static bond relations, of the Lennard-Jones potential, and of the reactive potential model due to Brenner. First, the contribution of the united-atom potential given in equation (76) can be straightforwardly calculated with a complexity of linear order, because for the case of a linear chain with static bond relations, the number of elements in the set of pairs $\mathcal{B}$, the set of triples $\mathcal{A}$, and the set of quadruples $\mathcal{T}$ is only of the order the number of particles. Second, to compute the contribution of the Lennard-Jones potential and of the Brenner potential given in equation (67), we have to evaluate sums over all pairs of particles and over all triples of particles due to the evaluation of the bond order terms B_{ij} in equation (72). Here, we can exploit that all involved contributions are of short range. We then obtain an implementation that scales only linear in the number of particles. In the following, we present some details of our implementation of Brenner's potential model, which is based on the linked cell technique [29, 92].

The idea of the linked cell method [176] is to decompose the simulation domain into equally sized cells. This way, the interactions of the particles for one cell are confined to the particles which at most belong to the neighboring cells. Let us first consider how to obtain an appropriate decomposition of the simulation domain for a given cutoff radius $r_c > 0$. To this end, we define the number of cells for the dth basis vector $\mathbf{a}_d$ in the form

$$n_{c_d} = \left\lfloor \frac{\left\|\mathbf{a}_d^{\perp}\right\|}{r_c} \right\rfloor ,$$

where $\mathbf{a}_d^{\perp}$ is given in equation (31) and $\lfloor x \rfloor$ denotes the maximal integer less than or equal to x. Then a particle i with scaled coordinates $\mathbf{s}_i$ is uniquely associated with a cell $C_{\mathbf{m}}$, if the condition

$$\frac{\mathbf{m}_d}{n_{c_d}} \leq \mathbf{s}_{i_d} < \frac{\mathbf{m}_d + 1}{n_{c_d}}$$

is fulfilled for all $d \in \{1, 2, 3\}$. Here, the indices $\mathbf{m}$ are in $\mathbb{N}^3$ with $0 \leq \mathbf{m}_d < n_{c_d}$. Thus, the domain of a cell $C_{\mathbf{m}}$ is given by the staggered span of the vectors

$$\mathbf{l}_{c_1} = \frac{\mathbf{a}_1}{n_{c_1}} , \qquad \mathbf{l}_{c_2} = \frac{\mathbf{a}_2}{n_{c_2}} , \qquad \mathbf{l}_{c_3} = \frac{\mathbf{a}_3}{n_{c_3}} .$$

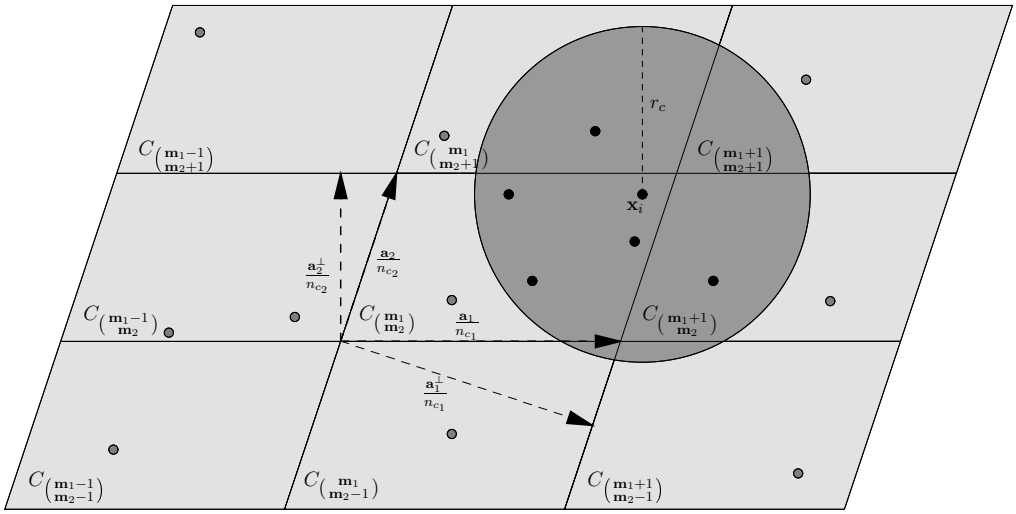


Figure 8: A cell $C_{\mathbf{m}}$ and their neighboring cells related to a decomposition of the simulation cell with $h = [\mathbf{a}_1, \mathbf{a}_2]$ for a cutoff radius r_c in two dimensions.

Note that the norm $\left\| \frac{\mathbf{a}_d^\perp}{n_{c_d}} \right\|$ is greater than or equal to the cutoff radius r_c. Thus, for each particle i associated with a cell $C_{\mathbf{m}}$, all particles j with a distance r_{ij} less than r_c are associated to the cell $C_{\mathbf{m}}$ or one of its 26 neighboring cells; see Figure 8. Note that in the case of an orthogonal simulation cell, the vectors $\mathbf{a}_1^\perp$, $\mathbf{a}_2^\perp$, and $\mathbf{a}_3^\perp$ are equal to the basis vectors $\mathbf{a}_1$, $\mathbf{a}_2$, and $\mathbf{a}_3$. Consequently, the equations $n_{c_d} = \left\lfloor \frac{\|\mathbf{a}_d\|}{r_c} \right\rfloor = \left\lfloor \frac{\|\mathbf{a}_d^\perp\|}{r_c} \right\rfloor$ and $l_{c_d} := \left\| \frac{\mathbf{a}_d}{n_{c_d}} \right\| = \left\| \frac{\mathbf{a}_d^\perp}{n_{c_d}} \right\|$ hold for $d = 1, 2, 3$; see Figure 9. For example, if we assume an approximately constant particle density ρ, we then can compute the contribution of an pair potentials u^{ij} with cutoff radius r_c with the complexity of the order $\mathcal{O}(\rho^2 N)$ in the form

$$U = \frac{1}{2} \sum_{\mathbf{m}_1=0}^{n_{c_1}-1} \sum_{\mathbf{m}_2=0}^{n_{c_2}-1} \sum_{\mathbf{m}_3=0}^{n_{c_3}-1} \sum_{i \in \mathcal{C}_{\mathbf{m}}} \sum_{\mathbf{m}' \in \mathcal{N}_c} \sum_{\substack{j \in \mathcal{C}_{\mathbf{m}+\mathbf{m}'} \\ j \neq i}} u^{ij}(r_{ij}) \,, \tag{78}$$

where $\mathcal{C}_{\mathbf{m}}$ denotes the set of particles i which are associated to cell $C_{\mathbf{m}}$ and the set of offsets $\mathcal{N}_c$ is given by

$$\mathcal{N}_c = \left\{ \begin{pmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{pmatrix} : \mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3 \in \{-1, 0, 1\} \right\} .$$

Note that in the case of periodic boundary conditions, the sum $\mathbf{m} + \mathbf{m}'$ in equation (78) is defined as

$$\begin{pmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \mathbf{m}_3 \end{pmatrix} + \begin{pmatrix} \mathbf{m}'_1 \\ \mathbf{m}'_2 \\ \mathbf{m}'_3 \end{pmatrix} := \begin{pmatrix} (\mathbf{m}_1 + \mathbf{m}'_1) \bmod n_{c_1} \\ (\mathbf{m}_2 + \mathbf{m}'_2) \bmod n_{c_2} \\ (\mathbf{m}_3 + \mathbf{m}'_3) \bmod n_{c_3} \end{pmatrix} .$$

Analogously, we can compute the forces with the complexity of the order $\mathcal{O}(\rho^2 N)$. In Algorithm 1 we give the basic structure of a linked cell algorithm to compute the

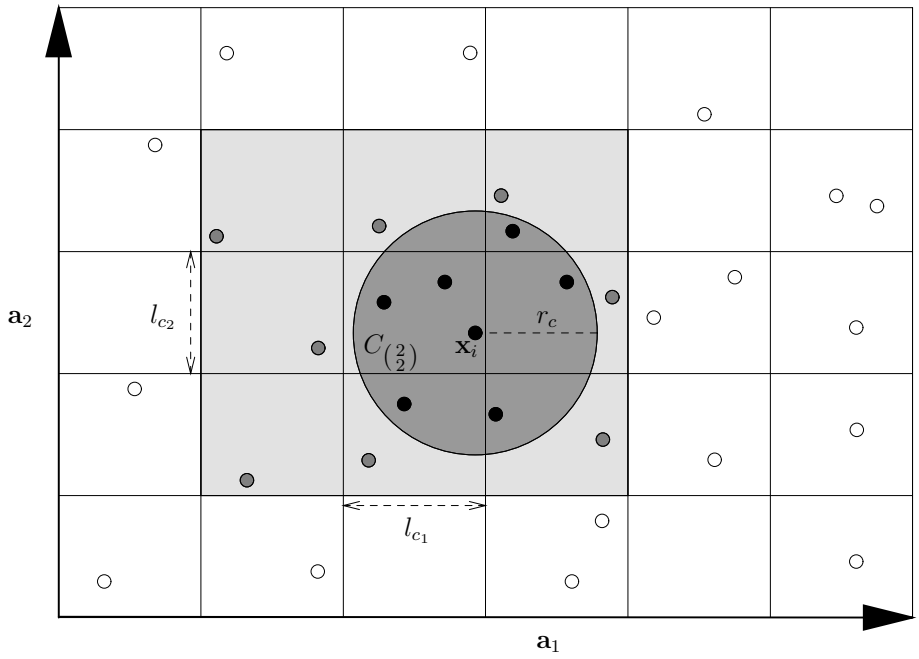


Figure 9: Decomposition related to a cutoff radius r_c of an orthogonal simulation cell with $h = [\mathbf{a}_1, \mathbf{a}_2]$ in two dimensions. To obtain for all particles i of a cell $C_{\mathbf{m}}$ all particles with r_{ij} less than r_c, we only have to look over the particles associated with the shaded cells.

Algorithm 1 Linked cell loop analog to equation (78) for the potential energy and force evaluation in three dimensions in the case of a pair potential. Note that the algorithm has to start with $U = 0$.

```
for all offsets m such that 0 ≤ m_d < n_{c_d} for each d = 1, 2, 3 do
  for all particles i ∈ C_m do
    set F_i := 0
    for all offsets m' ∈ N_c do
      for all particles j ∈ C_{m+m'} do
        evaluate r_ij
        if r_ij < r_c then
          evaluate u^{ij}(r_ij) and add it to U
          evaluate F_ij and add it to F_i
        end if
      end for
    end for
  end for
end for
```

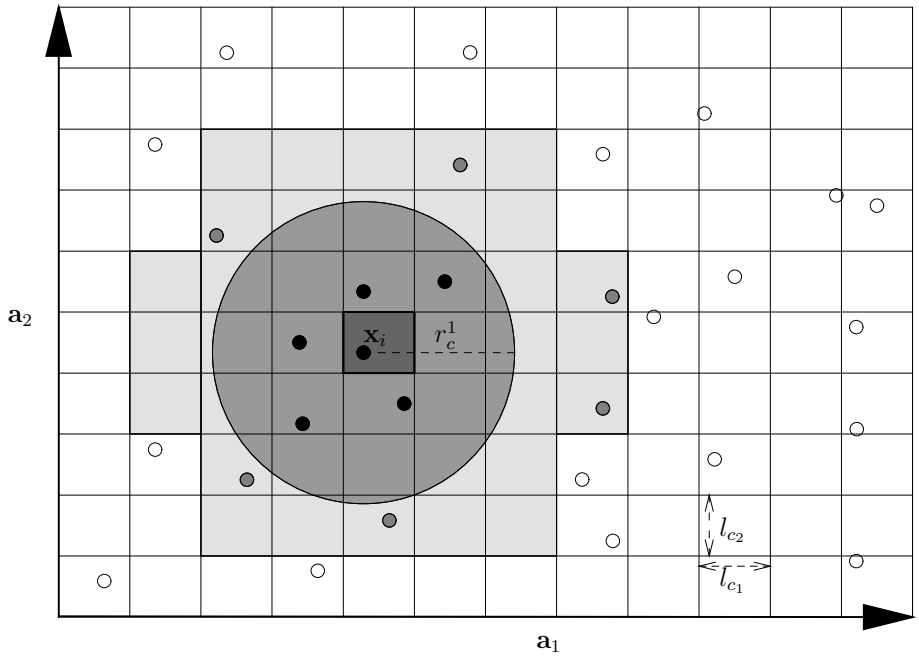


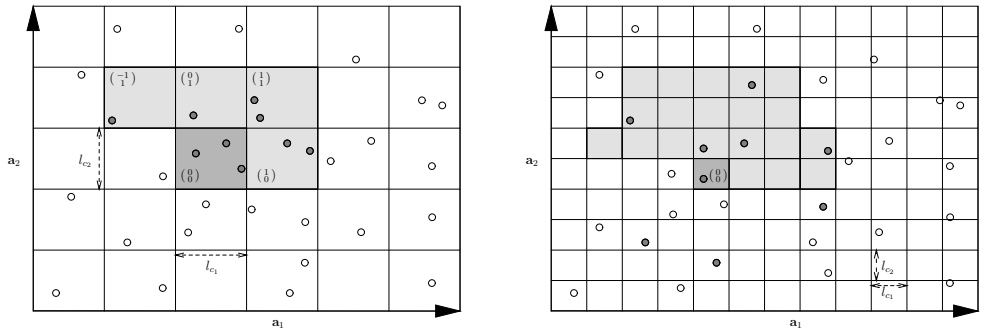
Figure 10: Decomposition related to a cutoff radius r_c of an orthogonal simulation cell with $h = [\mathbf{a}_1, \mathbf{a}_2]$ in two dimensions. To obtain all particles j with r_{ij} less than r_c^1 for all particles i of a cell $C_{\mathbf{m}}$, we only have to look for the particles associated with the shaded cells. The offset set $\mathcal{N}_c^1$ for the cutoff radius r_c^1 corresponds to the offset of the shaded cells with respect to the dark-shaded cell.

potential energy and the forces for each particle. In the following we will refer to this kind of algorithm as *linked cell loop*. For a detailed implementation, an array or a linked list data structure can be used for the cells. The particles of each cell with their associated data are stored in a linked list. Note that we have to update the linked cell structure in each timestep according to the cell matrix $h(t)$ and the actual particle distribution.

Let us now consider how to deal with several different cutoff radii $r_c^1 < r_c^2 < \cdots < r_c^{l_{\max}}$. Obviously, one possibility is to create one linked cell structure for the largest cutoff radius $r_c^{l_{\max}}$. But because the number of particles in a cell depends on the cutoff radius in third order, the resulting cost constant can be relatively large. Another possibility is to create one linked cell structure for *each* cutoff radius r_c^l. This however involves addtional storage and more complicated programming. Therefore, we proceed as follows: We use only one linked cell structure with an appropriate cutoff radius r_c but now take more than just next-neighbor cells into account. To this end, we have to generate an appropriate offset set $\mathcal{N}_c^l$ for each cutoff radius r_c^l; see Figure 10. Here, the sets $\mathcal{N}_c^l$ can be defined by

$$N_c^l := \left\{ \mathbf{m}' \in \mathbb{Z}^3 \,:\, \mathrm{dist}\,(C_{\mathbf{0}}, C_{\mathbf{m}'}) \leq r_c^l \right\} .$$

Now we can compute the contribution of pair potentials $u_1^{ij}, u_2^{ij}, \ldots, u_{l_{\max}}^{ij}$ with cutoff



(a) (b)

Figure 11: Decomposition related to a cutoff radius r_c of an orthogonal simulation cell with $h = [\mathbf{a}_1, \mathbf{a}_2]$ in two dimensions. (a) The offset set $\tilde{\mathcal{N}}_c$ in the case of a symmetric pair potential for the cutoff radius r_c corresponds to the offsets of the shaded cells with respect to the dark-shaded cell. (b) The offset set $\tilde{\mathcal{N}}_c^1$ in the case of a symmetric pair potential for the cutoff radius r_c^1 corresponds to the offsets of the shaded cells with respect to the dark-shaded cell.

radii $r_c^1 < r_c^2 < \cdots < r_c^{l_{\max}}$ in the form

$$U = \frac{1}{2} \sum_{\mathbf{m}_1=0}^{n_{c_1}-1} \sum_{\mathbf{m}_2=0}^{n_{c_2}-1} \sum_{\mathbf{m}_3=0}^{n_{c_3}-1} \sum_{i \in \mathcal{C}_{\mathbf{m}}} \sum_{l=1}^{l_{\max}} \sum_{\mathbf{m}' \in \mathcal{N}_c^l} \sum_{\substack{j \in \mathcal{C}_{\mathbf{m}+\mathbf{m}'} \\ j \neq i}} u_l^{ij}(r_{ij}) \; . \tag{79}$$

Note that in the case of $u_l^{ij} \equiv u_l^{ji}$, and thus $\mathbf{F}_{ij} = -\mathbf{F}_{ji}$, the computational cost can be reduced by the factor of two. To this end, the offset sets $\mathcal{N}_c^l$ can be adapted such that for each triple (i, j, l) either $u_l^{ij}(r_{ij})$ or $u_l^{ji}(r_{ij})$ is taken into account only; see Figure 11. To this end, we can define a set of offsets $\tilde{\mathcal{N}}_c^l$ as the maximal subset of $\mathcal{N}_c^l$ which fulfills the following conditions:

1. $\mathbf{0}$ is in $\tilde{\mathcal{N}}_c^l$.
2. If $\mathbf{m} \neq \mathbf{0}$ is in $\tilde{\mathcal{N}}_c^l$, then $-\mathbf{m}$ is not in $\tilde{\mathcal{N}}_c^l$.

Note that this definition of a set $\tilde{\mathcal{N}}_c^l$ is not unique, but we can choose just any and keep it fixed in a linked cell loop. Now, we can write equation (79) in the form

$$U = \sum_{\mathbf{m}_1=0}^{n_{c_1}-1} \sum_{\mathbf{m}_2=0}^{n_{c_2}-1} \sum_{\mathbf{m}_3=0}^{n_{c_3}-1} \sum_{i \in \mathcal{C}_{\mathbf{m}}} \sum_{l=1}^{l_{\max}} \left(\sum_{\mathbf{m}' \in \tilde{\mathcal{N}}_c^l \setminus \mathbf{0}} \sum_{j \in \mathcal{C}_{\mathbf{m}+\mathbf{m}'}} u_l^{ij}(r_{ij}) + \sum_{\substack{j \in \mathcal{C}_{\mathbf{m}} \\ j > i}} u_l^{ij}(r_{ij}) \right) , \tag{80}$$

where, with respect to offset $\mathbf{0}$, we still have to take either triple (i, j, l) or triple (j, i, l) into account. It is now straightforward to adapt the linked cell loop given in Algorithm 1 to equation (80) and to compute the potential energy as well as the forces this way.

Altogether, in our implementation concerning the potential model I given in equation (67) we follow the above-explained approach in the framework of the linked cell

technique. We use one linked cell structure that is associated to a cutoff radius $r_c = 2\,\text{Å}$. This way, we are able to compute all terms with respect to Brenner's potential model term u_B with the help of the associated offset sets $\mathcal{N}_c$ and $\tilde{\mathcal{N}}_c$; see Table 1. In addition, we generate an offset set $\tilde{\mathcal{N}}_c^W$ related to the cutoff radius $r_c^W = 10\,\text{Å}$ for the van der Waals term u_W; see Table 5. Then, we are able to evaluate the potential energy expression U_I in four linked cell loops:

1. Linked cell loop with offset set $\tilde{\mathcal{N}}_c$ to compute $N_i^{(\text{C})}$ and $N_i^{(\text{H})}$ for all $i \in \mathcal{C}$.
   (a) Evaluate $f_{ij}(r_{ij})$; see equation (70).
2. Linked cell loop with offset set $\tilde{\mathcal{N}}_c$ to compute $N_i^{(c)}$ defined by equation (81) for all $i \in \mathcal{C}$.
   (a) Evaluate $f_{ij}(r_{ij})F\left(N_j^{(t)} - f_{ij}(r_{ij})\right)$; see also equation (71).
3. Linked cell loop with offset set $\tilde{\mathcal{N}}_c$ to compute $\sum_{i=1}^{N}\sum_{\substack{j=1\\j>i}}^{N} u_B(r_{ij})$; see equation (67).
   (a) Evaluate B_{ij} by local loop around particle i with offset set $\mathcal{N}_c$; see equation (72).
   (b) Evaluate B_{ji} by local loop around particle j with offset set $\mathcal{N}_c$; see equation (72).
   (c) Evaluate $N_{ij}^{(c)}$ with help of $N_i^{(c)}$ and $N_j^{(c)}$ according equation (82).
   (d) Evaluate $\bar{B}_{ij}$; see equation (73).
   (e) Evaluate $u_B(r_{ij})$.
4. Linked cell loop with offset set $\tilde{\mathcal{N}}_c^W$ to compute $\sum_{i=1}^{N}\sum_{\substack{j=1\\j>i}}^{N} u_W(r_{ij})$; see equation (67).
   (a) Evaluate $u_W(r_{ij})$; see equation (74).

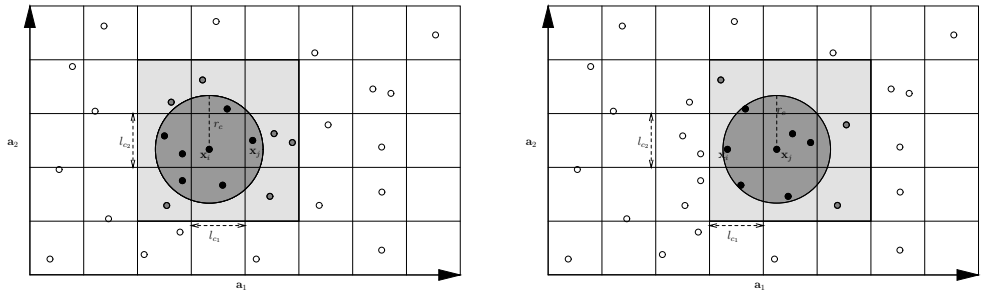
In the first linked cell loop, we compute the values $N_i^{(\text{C})}$ and $N_i^{(\text{H})}$ for all particles $i \in \mathcal{C}$ according to equation (70). With help of the values $N_i^{(\text{C})}$ and $N_i^{(\text{H})}$ we calculate the values

$$N_i^{(c)} := \sum_{\substack{j\in C\\ j\neq i}} f_{ij}(r_{ij})F\left(N_j^{(t)} - f_{ij}(r_{ij})\right) \tag{81}$$

for all particles $i \in \mathcal{C}$ in the next linked cell loop. This way, we are able to compute the values $N_{ij}^{(c)}$ given in equation (81) on the fly by the expression

$$N_{ij}^{(c)} = 1 + N_i^{(c)} - f_{ij}(r_{ij})F\left(N_j^{(t)} - f_{ij}(r_{ij})\right) + N_j^{(c)} - f_{ji}(r_{ji})F\left(N_i^{(t)} - f_{ji}(r_{ji})\right). \tag{82}$$

In the third linked cell loop that uses the offset set $\tilde{\mathcal{N}}_c$, we then compute the contribution of Brenner's potential. To this end, for each unique pair of particles i and j



(a) (b)

Figure 12: Decomposition related to a cutoff radius r_c of an orthogonal simulation cell with $h = [\mathbf{a}_1, \mathbf{a}_2]$ in two dimensions. Computation of expressions B_{ij} and B_{ji} for a pair of particles i and j, which is taken into account in the third linked cell loop to evaluate potential model I. (a) Local loop around particle i with offset set $\mathcal{N}_c$ to calculate B_{ij}. (b) Local loop around particle j with offset set $\mathcal{N}_c$ to calculate B_{ji}.

that has to be taken into account, we perform a local loop with respect to the offset set $\mathcal{N}_c$ around particle i to evaluate expression B_{ij} and a local loop around particle j to evaluate expression B_{ji}; see also Figure 12. Note that the sum over all particles k with $k \neq i$ and $k \neq j$ in the expression B_{ij} is restricted to particles k with a distance r_{ik} less than r_c due to the factor $f_{ik}(r_{ik})$; see equation (72). Now we evaluate the bond order term $\bar{B}_{ij}$ given in equation (73) with the help of the values B_{ij}, B_{ji} and $N_{ij}^{(c)}$. Then, we compute the contribution due to the van der Waals term u_W^{ij} with the help of the offset set $\tilde{\mathcal{N}}_c^1$ in the fourth linked cell loop. Thus we obtain the potential energy U_{I} given in equation (67). The force evaluation can be incorporated into our scheme in a straightforward way. We again use the linked cell technique and exploit symmetries accordingly. For more details we refer to [29].

Altogether, if we assume an approximately constant particle density ρ, our implementation for the potential models I and II, results in a complexity of the order $\mathcal{O}(\rho^3 N)$. The ρ^3-term stems here from the local loops to compute the expressions B_{ij} and B_{ji} within the third linked cell loop.

2.2.4 Parallel Implementation

A meaningful simulation of polymer-carbon nanotube composites involves a very large number of particles and long simulation times. Therefore a further acceleration of our code is necessary. This can be achieved by using a parallel computer system. To this end, our code must be parallelized properly. In the following, we give the basis of the parallelization of our linked cell method for parallel computers with distributed memory. We follow the classical domain decomposition approach [177, 92]. To this end, we assume that we have for each number of cells n_{c_d} a partition $(n_{c_d}^1, n_{c_d}^2, \ldots, n_{c_d}^{n_{p_d}})$ with $n_{c_d}^1 + n_{c_d}^2 + \cdots + n_{c_d}^{n_{p_d}} = n_{c_d}$. Now we obtain a decomposition of the simulation domain into $n_{p_1} \times n_{p_2} \times n_{p_3}$ subdomains $D_{\mathbf{M}}$ by

$$D_{\mathbf{M}} := \left\{ C_{\mathbf{m}} \; : \; \sum_{i=0}^{\mathbf{M}_d - 1} n_{c_d}^{i+1} \leq \mathbf{m}_d < \sum_{i=0}^{\mathbf{M}_d} n_{c_d}^{i+1} \, , \;\; d = 1, 2, 3 \right\} ,$$

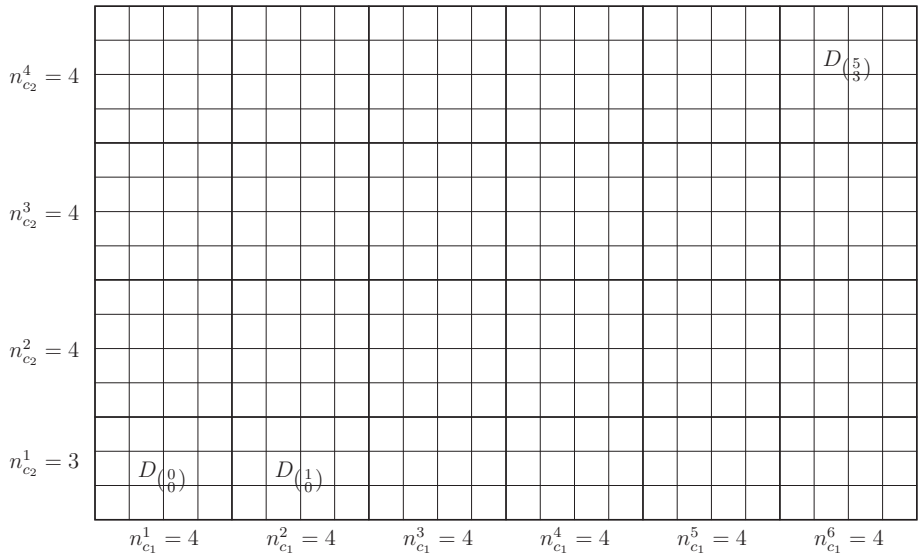


Figure 13: Decomposition of a two-dimensional simulation domain into 6×4 subdomains $D_{\mathbf{M}}$ by a partition of the 24×15 cells $C_{\mathbf{m}}$.

where the indices $\mathbf{M}$ are in $\mathbb{N}^3$ with $0 \leq \mathbf{M}_d < n_{p_d}$ for $d = 1, 2, 3$; see Figure 13. Then, each subdomain and its associated data is assigned to a processor. In principle, each processor can now work in parallel. However, during the computation, data of some cells are needed that belong to other processors. We therefore have to take special care for the appropriate handling and communication of such necessary data. To this end, additional layers of boundary cells, so-called *ghost cells*, are stored on each processor; see Figure 14. In these ghost cells, copies of the associated particles and their data are kept. In a first communication step, we exchange data between the processors such that the ghost cells of each processor store copies of the data that are associated to the layer cells of the neighboring subdomains adjacent to the boundary; see Figure 15. Note that the number of boundary layers n_{b_d} for each direction has to be chosen with respect to the offset sets used. In the case of the offset sets $\mathcal{N}_c^1, \ldots, \mathcal{N}_c^{l_{\max}}$, the number of boundary layers for the dth direction can be defined as

$$n_{b_d} := \max_{l=1,\ldots,l_{\max}} \max_{\mathbf{m} \in \mathcal{N}_c^l} \{|\mathbf{m}_d|\} \ .$$

This way, the size of the boundary related to the ghost cells is larger than the cutoff radius $r_c^{l_{\max}}$ in each direction. Thus, due to the copies of particle data in the ghost cells, the different processors now can compute their part of the linked cell loops fully in parallel. Then, depending on the details of the linked cell loop, communication of data is necessary to update the values of the particle data located in the boundary ghost cells; see Figure 16. Finally, additional communication is necessary in each time step after the positions of the particles are changed in the time integration scheme. Here, particles can move out of a subdomain into a neighboring one. Under the assumption that particles can move at most to their next neighboring cells, we can use the boundary ghost cells again to exchange these particles between the processors; see also Figure 16. The exchange of data is done in a synchronized, bidirectional fashion,

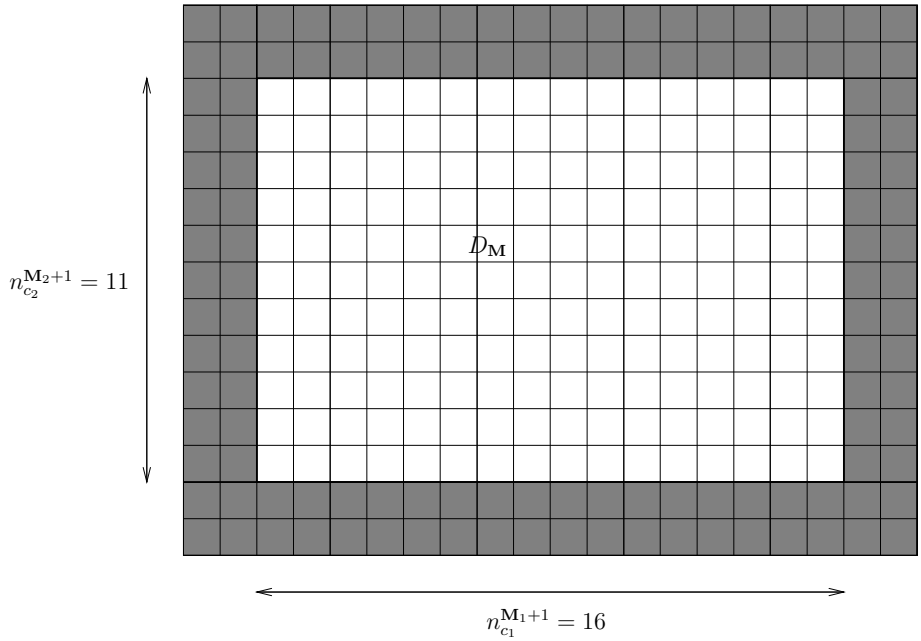


Figure 14: The subdomain $D_{\mathbf{M}}$ corresponds to the 16×11 unfilled cells. The additional ghost cells are shaded. In this two-dimensional example, the boundary consists of $n_{b_d} = 2$ layers of cells for $d = 1, 2$.

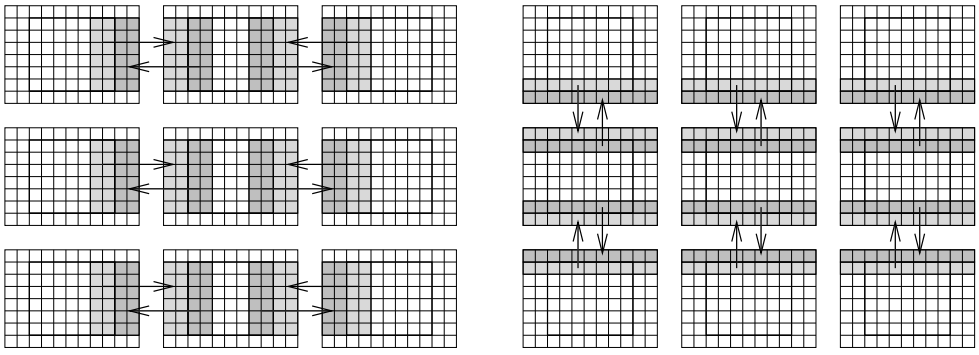


Figure 15: The communication pattern necessary to exchange and store copies of particle data in the boundary ghost cells can be performed in two steps in the two-dimensional case. Here, the number of boundary layers in the first direction is $n_{b_1} = 2$ and in the second direction it is $n_{b_2} = 1$. Only the communication pattern for one subdomain and its neighbors is depicted.

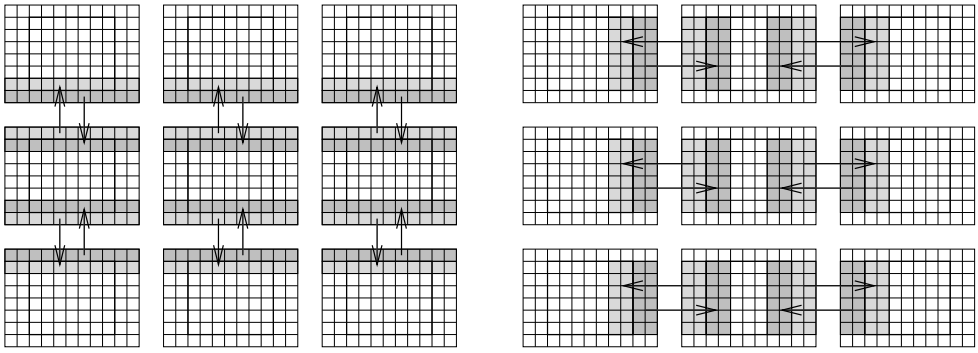


Figure 16: The communication to exchange data, which is situated in the boundary ghost cells can be performed in two steps in the two-dimensional case. In particular, its conjugated to the communication scheme represented in Figure 15. Only the communication for a subdomain and its neighbors is depicted.

parallel between subdomains in d consecutive steps, one step for each coordinate direction. Obviously, such communication of data is not performed cell by cell but collectively for all cells which belong to one side or face of each subdomain in one communication step; compare Figure 15 and Figure 16.

Altogether, our approach directly follows the well-known techniques from conventional domain decomposition parallelization [177, 178, 92]. The presented communication schemes can easily be adapted to the use of offset sets $\tilde{\mathcal{N}}_c$, which take advantage of symmetries. Also, the modifications necessary due to Brenner's potential are straightforward. Our implementation results in an optimal parallel complexity of the order $\mathcal{O}\left(\rho^3 \frac{N}{P}\right)$, where P denotes the number of processors. The increase of performance by using a parallel computer can be analyzed by the so-called *speedup*

$$\text{speedup}(P) := \frac{\text{runtime}(1)}{\text{runtime}(P)}$$

and the so-called *parallel efficiency*

$$\text{efficiency}(P) := \frac{\text{speedup}(P)}{P}\,,$$

where runtime(P) denotes the time of computation on P processors. Here, the optimal speedup is linear and the optimal parallel efficiency is one. We performed simulations with 320000 carbon atoms on a CRAY T3E-1200 to demonstrate the speedup and parallel efficiency of our parallel implementation. Our results show an almost linear speedup and a parallel efficiency of almost one; see Figure 17.

2.3 Elastic Moduli and Constants

The *stress-strain* relationship provides the overall mechanical response of a material when subjected to mechanical loading under certain conditions. One method to generate stress-strain curves is to vary the strain and to measure the stress in the framework of an (N, V, E) ensemble [37, 38, 39]. Alternatively, we can employ our

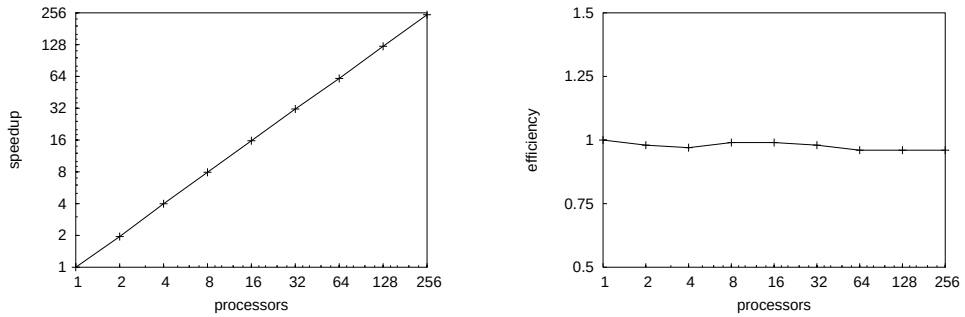


Figure 17: Speedup and parallel efficiency for up to 256 processors on a CRAY T3E-1200.

(N, P, T) ensemble approach to apply external stress and to measure the corresponding strain. For this purpose, we use an additional external stress tensor Π_{ext} within the equation of motion (22)

$$\dot{p}_h = \left(\Pi_{\text{int}} - \text{diag}\left(P_{\text{ext}}\right) + \Pi_{\text{ext}}\right) h^{-T} \det h - \frac{p_\eta}{M_T} p_h \,. \tag{83}$$

This way, we are able to accomplish various tensile and compressive load cases to study the elastic properties of a nanotube-polyethylene composite. To generate a stress-strain curve for a tensile or compressive load at given temperature and pressure, we first equilibrate the system with no external stress applied. We then increase or decrease the external stress over a period of time and measure the induced stress $\pi := -\Pi_{\text{int}}$ and the induced strain. To determine the strain, we express the instantaneous cell matrix

$$h = (1 + e) h_{\text{equi}}$$

in terms of the unique displacement matrix e and the equilibrated cell matrix h_{equi}. With the displacement defined as $\mathbf{u}(\mathbf{v}) = e\mathbf{v}$, the linear strain tensor ε is given as

$$\varepsilon_{\alpha\beta} = \frac{1}{2}\left(\frac{\partial u_\alpha}{\partial \mathbf{v}_\beta} + \frac{\partial u_\beta}{\partial \mathbf{v}_\alpha}\right) ,$$

which in particular equals the symmetric part $\frac{1}{2}\left(e^T + e\right)$ of the displacement matrix $e = h h_{\text{equi}}^{-1} - 1$. The skew-symmetric part $\frac{1}{2}\left(e^T - e\right)$ of e corresponds to the so-called *linear rotational strain tensor*. Because the rotations of the unit cell do not convey any physical meaning, only six (instead of nine; see section 2.1) degrees of freedom are required to control the pressure within an (N, P, T) ensemble. Therefore, we constrain the 3×3 matrix $\dot{p}_h$ in the equation of motion (83) to be symmetric. We further assume a symmetric cell matrix h_{equi}. Then, the linear rotational strain tensor vanishes, and the linear strain tensor ε equals the displacement matrix e.

In particular, we want to determine *elastic constants*. To calculate the components of the elasticity tensor, the so-called elastic moduli $C_{\alpha\beta\gamma\delta}$, we use the *generalized Hooke's law* [179]

$$\pi_{\alpha\beta} = \sum_{\gamma\delta} C_{\alpha\beta\gamma\delta} \varepsilon_{\gamma\delta} \,, \quad \alpha, \beta \in \{1, 2, 3\} \,. \tag{84}$$

Here, we assume that $\pi_{\alpha\beta}$ are homogeneous linear functions of $\varepsilon_{\gamma\delta}$ and that C is a positive-definite fourth-order tensor that obeys the *major symmetry* $C_{\alpha\beta\gamma\delta} = C_{\gamma\delta\alpha\beta}$ [180]. For most solid materials, such relations hold until the stress reaches the so-called proportional limit. If the stresses exceed this limit, the deformation becomes nonlinear, but the elastic behavior of the material continues until the stresses reach the so-called elastic limit. The general fourth-order tensor C has $3^4 = 81$ independent constants $C_{\alpha\beta\gamma\delta}$. But because π and ε are symmetric second-order tensors, the number of independent elastic moduli reduces to $6 \times 6 = 36$. Because of the major symmetry, it further reduces to $6(6+1)/2 = 21$. Then, we can write the stress-strain relation (84) in matrix form

$$\begin{pmatrix} \pi_{11} \\ \pi_{22} \\ \pi_{33} \\ \pi_{12} \\ \pi_{13} \\ \pi_{23} \end{pmatrix} = \begin{pmatrix} C_{1111} & C_{1122} & C_{1133} & C_{1112} & C_{1113} & C_{1123} \\ C_{1122} & C_{2222} & C_{2233} & C_{2212} & C_{2213} & C_{2223} \\ C_{1133} & C_{2233} & C_{3333} & C_{3312} & C_{3313} & C_{3323} \\ C_{1112} & C_{2212} & C_{3312} & C_{1212} & C_{1213} & C_{1223} \\ C_{1113} & C_{2213} & C_{3313} & C_{1213} & C_{1313} & C_{1323} \\ C_{1123} & C_{2223} & C_{3323} & C_{1323} & C_{1323} & C_{2323} \end{pmatrix} \begin{pmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{12} \\ 2\varepsilon_{13} \\ 2\varepsilon_{23} \end{pmatrix} . \tag{85}$$

We can invert the stress-strain relations (85) by inverting the symmetric 6×6 elastic constant matrix C. This results in

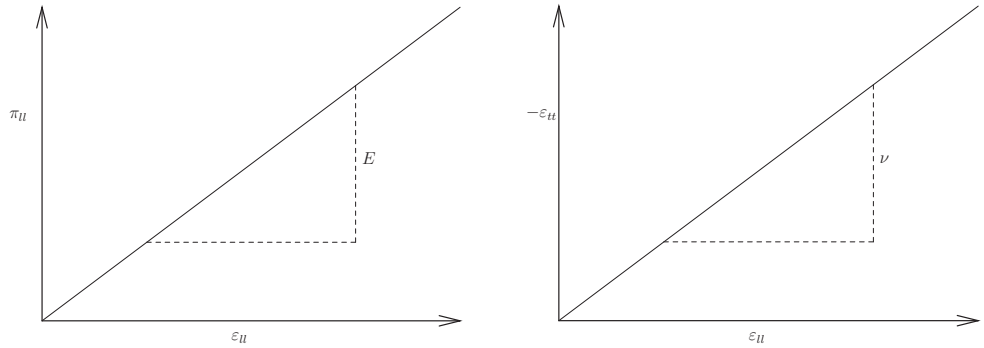
$$\begin{pmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ 2\varepsilon_{12} \\ 2\varepsilon_{13} \\ 2\varepsilon_{23} \end{pmatrix} = \begin{pmatrix} S_{1111} & S_{1122} & S_{1133} & S_{1112} & S_{1113} & S_{1123} \\ S_{1122} & S_{2222} & S_{2233} & S_{2212} & S_{2213} & S_{2223} \\ S_{1133} & S_{2233} & S_{3333} & S_{3312} & S_{3313} & S_{3323} \\ S_{1112} & S_{2212} & S_{3312} & S_{1212} & S_{1213} & S_{1223} \\ S_{1113} & S_{2213} & S_{3313} & S_{1213} & S_{1313} & S_{1323} \\ S_{1123} & S_{2223} & S_{3323} & S_{1323} & S_{1323} & S_{2323} \end{pmatrix} \begin{pmatrix} \pi_{11} \\ \pi_{22} \\ \pi_{33} \\ \pi_{12} \\ \pi_{13} \\ \pi_{23} \end{pmatrix} ,$$

where S is the so-called compliance matrix. Suppose now that only one of the six independent components of the induced stress is nonzero, say $\pi_{\alpha\beta}$. Then the components

$$S_{11\alpha\beta} = \frac{\varepsilon_{11}}{\pi_{\alpha\beta}} , \qquad S_{22\alpha\beta} = \frac{\varepsilon_{22}}{\pi_{\alpha\beta}} , \qquad S_{33\alpha\beta} = \frac{\varepsilon_{33}}{\pi_{\alpha\beta}} ,$$
$$S_{12\alpha\beta} = \frac{2\varepsilon_{12}}{\pi_{\alpha\beta}} , \qquad S_{13\alpha\beta} = \frac{2\varepsilon_{13}}{\pi_{\alpha\beta}} , \qquad S_{23\alpha\beta} = \frac{2\varepsilon_{23}}{\pi_{\alpha\beta}} ,$$

can be determined by calculating the slopes of the corresponding stress-strain curves. Here, the slopes can be easily computed by least squares linear regression [89]. This way, we successively determine the components $S_{\alpha\beta\gamma\delta}$ and consequently the elastic moduli $C_{\alpha\beta\gamma\delta}$.

There is a special elastic constant, the so-called *Young's modulus* E. If the stretching force is only applied in uniaxial direction, for example in longitudinal direction, the constant $E_{ll} := \pi_{ll}/\varepsilon_{ll} = 1/S_{llll}$ represents the ratio of the longitudinal stress to the corresponding longitudinal strain. In other words, E_{ll} is the slope of the stress-strain curve under uniaxial tension. Furthermore, the ratio of transverse contraction strain to longitudinal extension strain in the direction of the stretching force is known as the *Poisson ratio* $\nu := -\varepsilon_{tt}/\varepsilon_{ll}$; see Figure 18. Here, tensile deformation is considered positive and compressive deformation is considered negative. Thus, normal materials have a positive ratio. Furthermore, the generalized versions of the Poisson ratio are given by $\nu_{tt,ll} := -S_{ttll}/S_{llll}$.



(a) (b)

Figure 18: (a) The Young's modulus can be derived from the slope of the longitudinal stress versus longitudinal strain curve. (b) The Poisson ratio can be derived from the slope of the negative transversal strain versus longitudinal strain curve.

3 Numerical Experiments

In this section, we describe aspects and results of the numerical experiments we performed to study the mechanical properties of polyethylene-carbon nanotube composites. Here, we first give a summary of the systems we analyzed, where we outline the creation of the initial structures as well as the initial equilibration process. Then, we present the results of our tensile load test simulations.

We have incorporated the computational methods described in section 2 into our molecular dynamics software package TREMOLO, which is a load-balanced distributed memory parallel code [92]. For details and further information see the web page

http://www.ins.uni-bonn.de/research/projects/tremolo

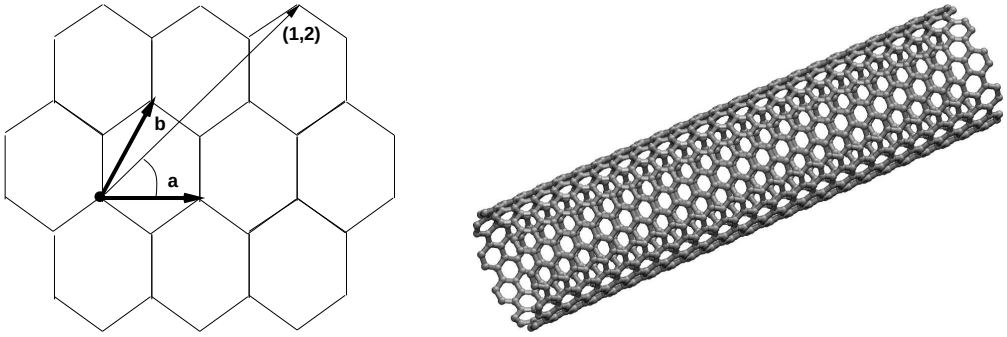
All numerical experiments were performed on our PC cluster Parnass2 [181]. This parallel computing system consists of 128 Intel Pentium II 400 MHz processors connected by a 1.28 GBit/s switched Myrinet.

3.1 Equilibrated Systems

Any molecular dynamics simulation should start with a relaxed, equilibrated system. Basically, we generated such equilibrated reference systems in three subsequent steps. First, we created an appropriate initial system, then we performed a nonlinear conjugated gradient scheme in order to relax the system to its local potential energy minimum. Finally, we conducted an isobaric-isothermal molecular dynamics simulation for 100.0 ps under normal conditions (i.e. for a temperature of 273.15 K and for an external pressure of $1.01325 \cdot 10^{-4}$ GPa). The fictitious mass M_T for the thermostat was set to 10.0 uÅ^2, and the fictitious mass M_P for the barostat was set to 10.0 u. We applied periodic boundary conditions for all systems.

3.1.1 Carbon Nanotubes

A single-walled carbon (n, m)-nanotube is a rolled-up sheet of graphene. The (n, m) integer indices correspond to the chiral vector along which the graphene sheet has



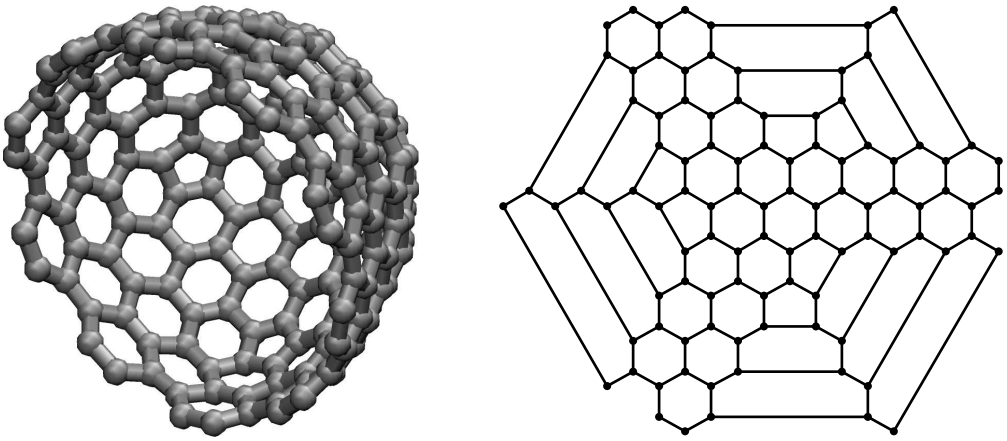
(a) (b)

Figure 19: (a) A strip of a graphite sheet is depicted. Here, the basis $\{\mathbf{a}, \mathbf{b}\}$ is given and an example of a roll-up vector with $(n, m) = (1, 2)$ is shown. The chiral angle θ is the angle between the vector $(1, 2)$ and the direction $\mathbf{a}$. (b) A $(10, 10)$ carbon nanotube with 960 atoms is depicted.

been rolled up. Here, the integer values (n, m) are given corresponding to the basis vectors $(\mathbf{a}, \mathbf{b})$. This way, their form depends on the so-called chiral angle θ between the roll-up vector (n, m) and the basis direction $\mathbf{a}$; see Figure 19. The tubule diameter δ and the chiral angle θ are uniquely determined by the defining roll-up vector (n, m). Here, we have $\delta = 0.78\sqrt{n^2 + nm + m^2}$ angstrom for the diameter and $\theta = \arctan\left(\sqrt{3}m/(m + 2n)\right)$ for the chiral angle [7]. This way, we generated the coordinates of two $(10, 10)$ nanotubes, one with a length about 57 Å and another which is about 101 Å long; see Figure 19(b) and Figure 21(a). To obtain a capped nanotube we follow the approach of Asimov and Buja [182]. Here, six disjoint 60-degree wedges are removed from a graphene sheet, where each wedge is centered at a point of the hexagonal carbon lattice. Then, opposite boundary points of the intersections are defined to be adjacent; compare Figure 20. Finally, we have put two caps and the shorter $(10, 10)$ nanotube together and thus obtained the coordinates of a capped single walled nanotube; see Figure 21(b). To equilibrate the capped nanotube, we fixed the size of the simulation cell. In the case of the uncapped nanotube, we applied appropriate constraints in the respective Lagrangian to permit the unit cell to alter in longitudinal direction only. Note that the uncapped nanotube spans the length of the periodic unit cell in the longitudinal direction. In the following, we will denote this system as *UC*.

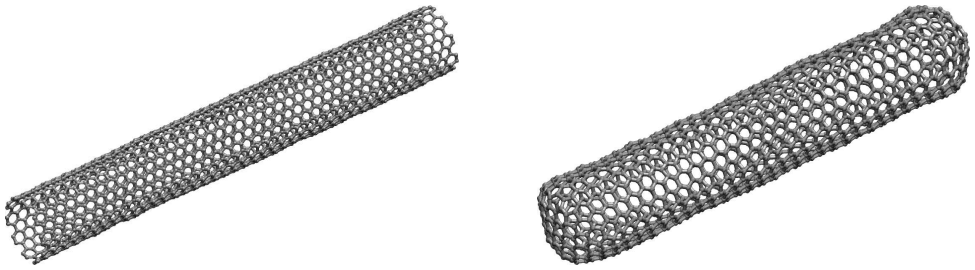
3.1.2 Polyethylene Matrices

To create the initial coordinates of a periodically replicated polyethylene matrix with N_{chain} chains of length L_{chain}, we randomly distributed N_{chain} monomeric $\tilde{\text{C}} = \text{CH}_2$ units in an appropriate simulation domain. Starting from these initial $\tilde{\text{C}}$s, we built up the polyethylene chains by successively adding new $\tilde{\text{C}}$ units until the length L_{chain} was reached. Here, the coordinates of the newly introduced $\tilde{\text{C}}$'s were randomly generated. To this end, we used the fixed bond distance of a $\tilde{\text{C}}$-$\tilde{\text{C}}$ bond but a randomly chosen bond angle between 90% and 110% for the $\tilde{\text{C}}$-$\tilde{\text{C}}$-$\tilde{\text{C}}$ bond angle. Also, we randomly chose the torsional angle; compare also Table 6. We then performed a local minimization on the system and finally run an isobaric-isothermal molecular



(a) (b)

Figure 20: (a) A cap for a $(10, 10)$ carbon nanotube with 188 atoms is depicted. (b) It is derived from a modified graphite sheet; see also [182].



(a) (b)

Figure 21: (a) A continuous $(10, 10)$ carbon nanotube with 1680 atoms at a temperature of 300 K. (b) A capped $(10, 10)$ carbon nanotube with 1136 atoms and a length about 67.5 Å at a temperature of 300 K.

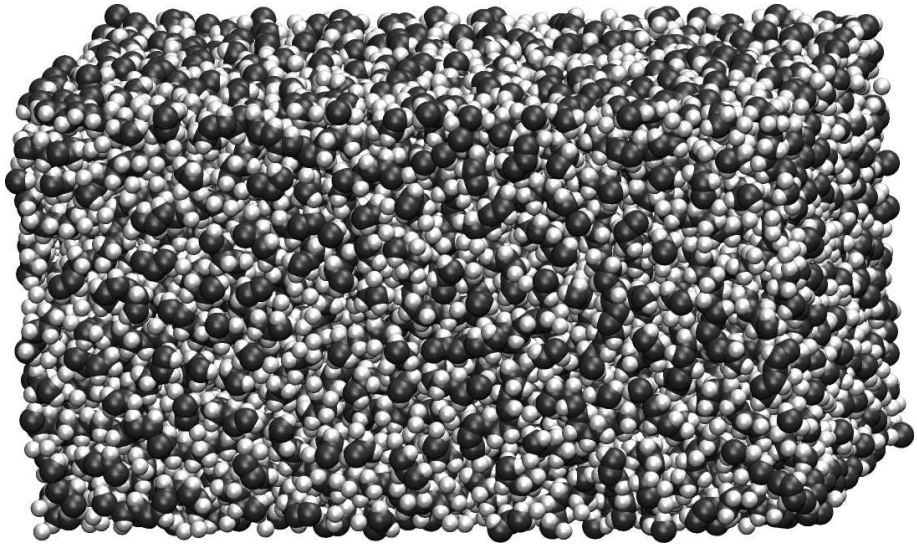


Figure 22: Equilibrated periodic unit cell of a polyethylene matrix composed of 10 chains of length 1000 represented with model I (system MIa).

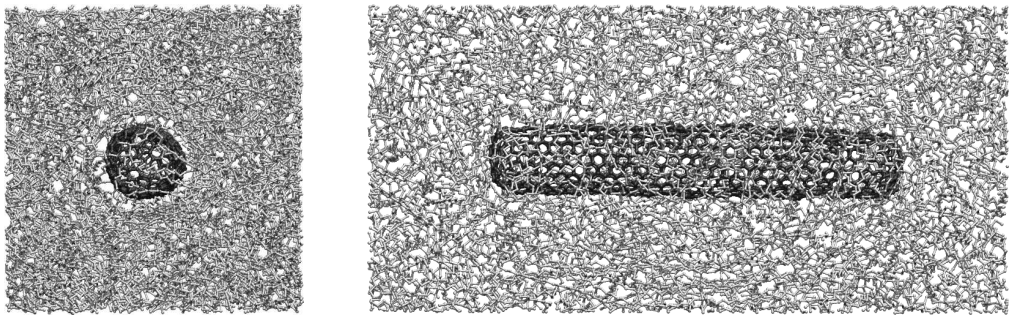
dynamics simulation under normal conditions. Note that with respect to model I, we used the coordinates of the monomeric units for the carbon atoms and added the remaining hydrogen atoms before the local minimization process. This way, we generated the following equilibrated polyethylene matrices:

- A polyethylene matrix, containing 10 chains of 1000 CH_2 units; see Figure 22. Here we used model I. We denote this system as *MIa*.
- A polyethylene matrix, containing 10 chains of 1000 monomeric $\tilde{C}$ units. Here we used model II. We denote this system as *MIIa*.
- A polyethylene matrix, containing 20 chains of 1000 monomeric $\tilde{C}$ units. Here we used model II. We denote this system as *MIIb*.
- A polyethylene matrix, containing 30 chains of 1000 monomeric $\tilde{C}$ units. Here we used model II. We denote this system as *MIIc*.

All equilibrated matrices exhibit a density of about $1\,g/cm^3$.

3.1.3 Composites

To model a polyethylene-carbon nanotube composite, we embedded a previously relaxed carbon nanotube into a previously equilibrated polyethylene matrix. Here, we followed Brown's approach [183] to create a cylindric cavity in the polyethylene matrix by using an artificial soft repulsive potential [184] within a local minimization process. We then placed the carbon nanotube in the cylindric cavity and performed a local minimization process again. Finally, we run an isothermal-isobaric molecular dynamics simulation to obtain an equilibrated system. This way, we equilibrated the following systems:



(a) (b)

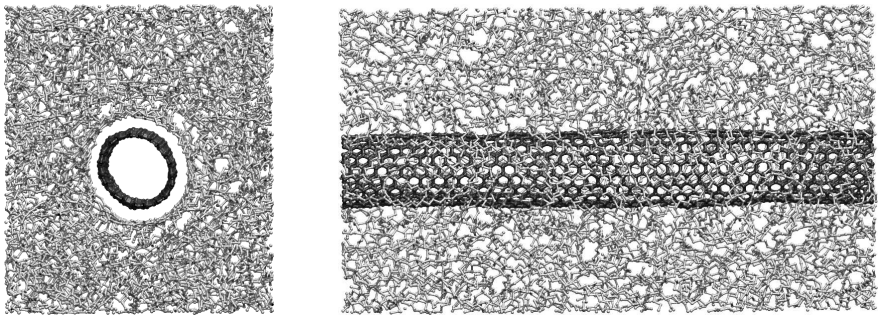
Figure 23: Equilibrated periodic unit cell of the capped nanotube in a polyethylene matrix represented with model II (system CIIb). (a) Front view of unit cell. (b) Side view of the unit cell.

Table 8: The diagonal components of the cell matrix of the equilibrated composite systems.

	CIa	CIIa	CIIb	CIIc	UCIIa
h_{11}[Å]	49.2	53.4	58.7	63.6	49.7
h_{22}[Å]	49.3	55.4	58.5	63.2	57.5
h_{33}[Å]	91.0	96.7	121.7	139.0	102.5

- The capped carbon nanotube embedded in matrix *MIa*. Here we used model I. We denote this system as *CIa*. The volume fraction of the nanotube is about 4.57 %.
- The capped carbon nanotube embedded in matrix *MIIa*; see Figure 23. Here we used model II. We denote this system as *CIIa*. The volume fraction of the nanotube is about 3.52 %.
- The capped carbon nanotube embedded in matrix *MIIb*. Here we used model II. We denote this system as *CIIb*. The volume fraction of the nanotube is about 2.41 %.
- The capped carbon nanotube embedded in matrix *MIIc*. Here we used model II. We denote this system as *CIIc*. The volume fraction of the nanotube is about 1.80 %.
- The periodically replicated uncapped carbon nanotube which spans the length of the unit cell and that is embedded in matrix *MIIa*; see Figure 24. Here we used model II. We denote this system as *UCIIa*. The volume fraction of the nanotube is about 5.22 %.

The diagonal components of the cell matrix of the equilibrated composite systems are given in Table 8.



(a) (b)

Figure 24: Equilibrated periodic unit cell of the periodically replicated uncapped nanotube in polyethylene represented with model II (system UCIIa). (a) Front view of unit cell. (b) Side view of the unit cell.

Figure 25: View of a functionalized carbon nanotube. Here, 1% of the carbon atoms have been randomly linked to a CH_2 unit.

3.1.4 Functionalized Nanotube Composites

The systems CIa, CIIa, CIIb, CIIc, and UCIIa are composites with weak nonbonded matrix nanotube interactions. One way to strengthen the interaction is with chemical cross-links. This is also known as functionalization [37, 38, 185]. To this end, we randomly placed CH_2 units between the matrix and the nanotube of the equilibrated system CIa. Then, we used the harmonic bond stretching potential u_{bond} of model II within a local minimization process to obtain cross-links between the matrix and the nanotube. Finally, we performed an isothermal-isobaric molecular dynamics simulation with the help of model I to obtain an equilibrated system. This way, we equilibrated the following three systems with different amounts of cross-links:

- A carbon nanotube embedded in the polyethylene matrix *MIa*, where 1% of the carbon atoms of the nanotube have been cross-linked; see Figure 25 and Figure 26. Here we used model I. We denote this system as *F1Ia*.
- A carbon nanotube embedded in matrix *MIa*, where 5% of the carbon atoms

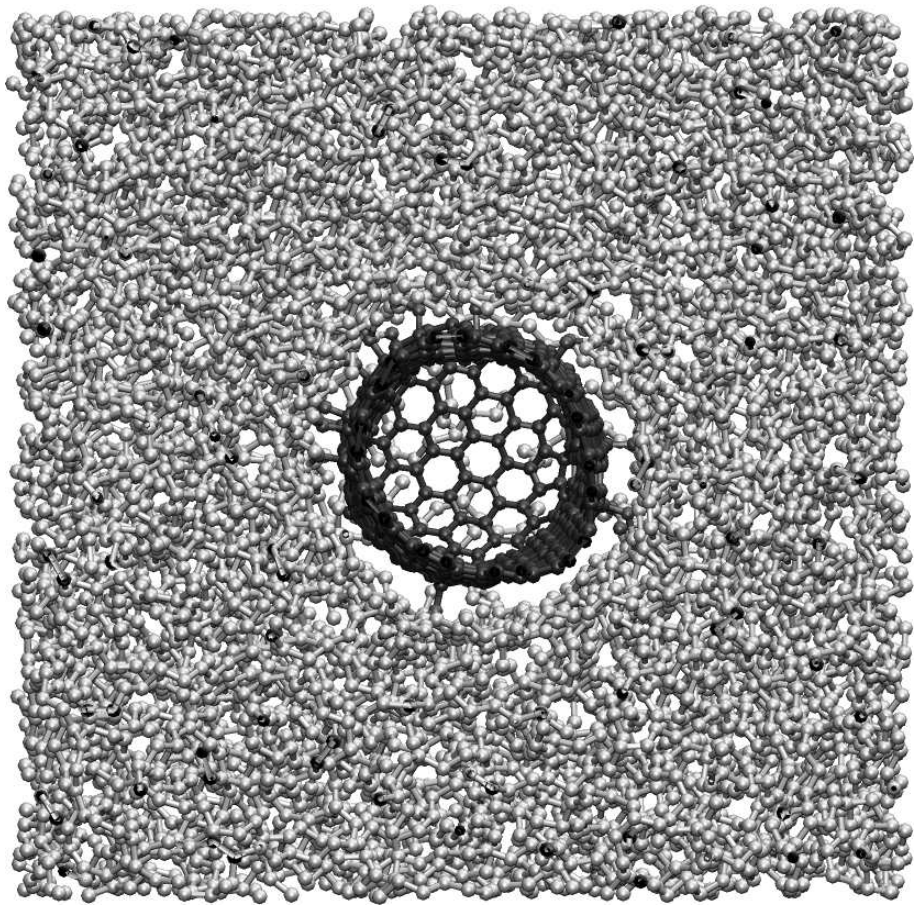


Figure 26: Equilibrated periodic unit cell of a nanotube in a polyethylene matrix represented with model II, where 1% of the carbon atoms of the nanotube have been cross-linked to the matrix. Note that we cut off the front part of the simulation domain and depict no hydrogen for better visualization.

of the nanotube have been cross-linked. Here we used model I. We denote this system as *F5Ia*.

- A carbon nanotube embedded in matrix *MIa*, where 10% of the carbon atoms of the nanotube are cross-linked. Here we used model I. We denote this system as *F10Ia*.

3.2 Simulation of Tensile Load Tests

All our tensile load test simulations were carried out under normal conditions. Here, the same time steps and fictitious masses as in the molecular dynamics part of the equilibration process were used. To apply a tensile load to one of the six independent stress components, we used a stress rate of 0.01 GPa/ps. A molecular dynamics tensile simulation was stopped when a strain of 10 % was reached.

Let us first consider the periodically replicated uncapped carbon nanotube (system UC). Here, we applied uniaxial external stress in the direction of the axis of the

nanotube to determine its Young's modulus and its Poisson ratio. For the modulus, we obtained a value of $E^{\mathrm{UC}} = 398.77\,\mathrm{GPa}$ and for the associated Poisson ratio we obtained a value of $\nu^{\mathrm{UC}} = 0.23$. Here, we assumed the carbon nanotube as a hollow cylinder with thickness 3.4 Å to calculate the volume [27]. This way, we obtained an equilibrated volume of $15.655\,\mathrm{nm}^3$. We also applied the Langrange strain tensor [179, 180] instead of the linear strain tensor and then obtained a Young's modulus of 395.83 GPa and a Poisson ratio of 0.22. If we applied the logarithmic strain tensor [179, 180] instead, we measured a Young's modulus of 401.72 GPa and a Poisson ratio of 0.24. Note that there is a lot of variance among the elastic moduli reported by several groups [186]. For example, about 300 GPa [187], 600 GPa [26, 28] and about 1 TPa [27] are given for the uniaxial Young's modulus of a $(10, 10)$ carbon nanotube in the literature. Also for the $(10, 10)$ nanotube, the Poisson ratio is noted as 0.25 [24], 0.278 [27], and 0.287 [187], respectively. The different approaches, models, and parameters used from the different groups are likely to be responsible for this variety of results.

We also simulated uniaxial tensile load tests for the matrices MIIa and MIIb. Here, we obtained a Young's modulus of about 2.1 GPa and a Poisson ratio of about 0.4. Note that the nanotubes are aligned to the third coordinate axis of the equilibrated composite systems. Thus, these systems are unidirectional composites and we therefore can assume that they are *orthotropic*. Then, the compliance matrix has only nine independent constants

$$\begin{pmatrix} S_{1111} & S_{1122} & S_{1133} & 0.0 & 0.0 & 0.0 \\ & S_{2222} & S_{2233} & 0.0 & 0.0 & 0.0 \\ & & S_{3333} & 0.0 & 0.0 & 0.0 \\ & & & S_{1212} & 0.0 & 0.0 \\ & & \mathrm{sym} & & S_{1313} & 0.0 \\ & & & & & S_{2323} \end{pmatrix},$$

because the basis vectors lie in the symmetry planes. To compute the compliance matrix and the elastic constant matrix for system UCIIa, we applied external stress to each of the six independent stress components. Then, the derived compliance matrix reads as

$$S^{\mathrm{UCIIa}} = \begin{pmatrix} 0.4990 & -0.1870 & -0.0109 & 0.0 & 0.0 & 0.0 \\ & 0.4904 & -0.0115 & 0.0 & 0.0 & 0.0 \\ & & 0.0428 & 0.0 & 0.0 & 0.0 \\ & & & 1.0440 & 0.0 & 0.0 \\ & \mathrm{sym} & & & 1.1135 & 0.0 \\ & & & & & 1.4728 \end{pmatrix}. \tag{86}$$

Now, by inverting the compliance matrix S^{UCIIa}, we obtained the elastic constant matrix in the form

$$C^{\mathrm{UCIIa}} = \begin{pmatrix} 2.3684 & 0.9232 & 0.8511 & 0.0 & 0.0 & 0.0 \\ & 2.4118 & 0.8852 & 0.0 & 0.0 & 0.0 \\ & & 23.7746 & 0.0 & 0.0 & 0.0 \\ & & & 0.9578 & 0.0 & 0.0 \\ & \mathrm{sym} & & & 0.8980 & 0.0 \\ & & & & & 0.67897 \end{pmatrix}. \tag{87}$$

Table 9: Elastic moduli and Poisson ratios of the composite systems that have been simulated using model II. The longitudinal moduli are increasing and the Poisson ratios are decreasing to the Poisson ratio of the $(10, 10)$ carbon nanotube $\nu^{\mathrm{UC}} = 0.23$.

System	E_{33}[GPa]	$\nu_{11,33}$	$\nu_{22,33}$	E_{11}[GPa]
UCIIa	23.3	0.25	0.26	2.00
CIIa	5.4	0.35	0.36	2.2
CIIb	3.8	0.38	0.37	
CIIc	3.0	0.39	0.39	

Table 10: Elastic moduli of the composite systems that have been simulated using model I. The longitudinal moduli are increasing with the amount of cross-links.

	MIa	CIa	F1Ia	F5Ia	F10Ia
E_{33}[GPa]	2.88	5.97	6.21	6.78	7.01

Here, we see a difference between the modulus $E_{33}^{\mathrm{UCIIa}} = \frac{1}{S_{33}} = 23.36\,\mathrm{GPa}$ and the moduli $E_{11}^{\mathrm{UCIIa}} = \frac{1}{S_{11}} = 2.00\,\mathrm{GPa}$ and $E_{22}^{\mathrm{UCIIa}} = \frac{1}{S_{22}} = 2.03\,\mathrm{GPa}$ due to the unidirectional reinforcement. We furthermore applied uniaxial stress to system CIIa in the direction of the first and third coordinate axis. Here, we obtained for the Young's modulus in longitudinal direction a value of $E_{33}^{\mathrm{CIIa}} = 5.4\,\mathrm{GPa}$ and in transversal direction a value of $E_{11}^{\mathrm{CIIa}} = 2.2\,\mathrm{GPa}$. In particular, the Young's moduli subjected to transversal loading conditions are in the same range as that of the matrix alone. Thus, we concentrate in the following on longitudinal loading conditions. For the other composite systems that are associated with model II, namely CIIb and CIIc, we applied uniaxial external stress to determine the Young's moduli and the Poisson ratios for the longitudinal direction. The results are summarized in Table 9.

In order to study the effect on the Young's modulus of cross-links between the nanotube and the matrix, we run simulations with uniaxial loads for the composite systems MIa, CIa, F1Ia, F5Ia, and F10Ia. Note that we used here model I to accurately describe the covalent bonds of the cross-links. The resulting Young's moduli for the longitudinal direction are given in Table 10; see also Figure 27.

4 Discussion

From the results of Table 9 we see the following: Subjected to *transversal* loading conditions, the Young's modulus of the composite is in the range of the modulus of the matrix. Thus, there is no reinforcement of the matrix. Subjected to *longitudinal* loading conditions, we see a Young's modulus that is 11.6 times higher for system UCIIa, 2.7 times higher for system CIIa, 1.9 times higher for system CIIb, and 1.5 times higher for system CIIc.

For a nanocomposite under uniaxial loading, the dependence of the elastic modulus on the nanotube volume fraction can be estimated by a macroscopic rule-of-

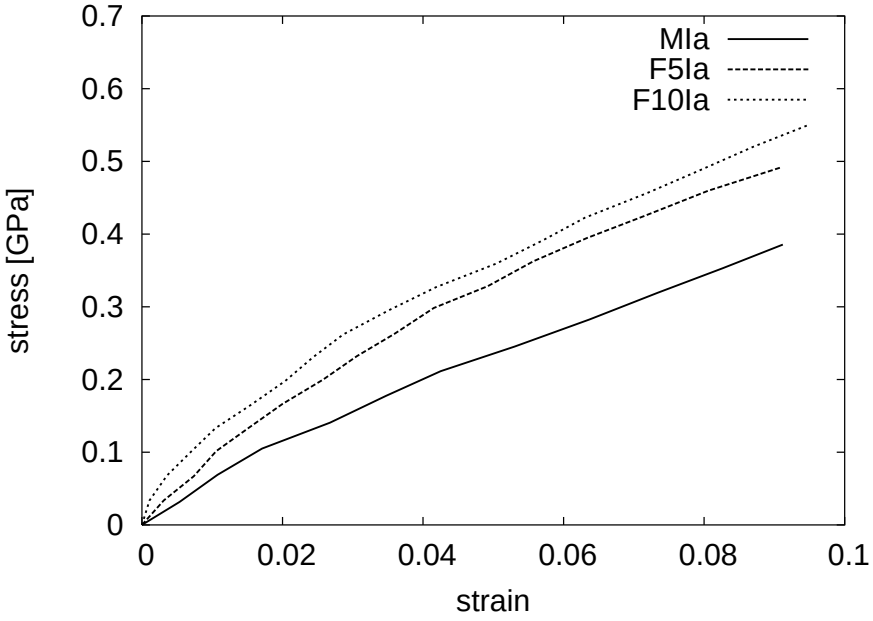


Figure 27: Stress-strain curves for systems MIa, F5Ia, and F10Ia.

mixtures. This rule reads in its simplest form as

$$E_c = V_f E_f + (1 - V_f) E_m \,, \tag{88}$$

where E_c denotes the predicted Young's modulus of the composite, E_f denotes the Young's modulus of the tube, E_m denotes the Young's modulus of the matrix and V_f denotes the volume fraction of the tube.

Let us consider system UCIIa first. The modulus we obtained for a $(10, 10)$ carbon nanotube is $E_f = 398.77\,\mathrm{GPa}$ and its volume fraction is $5.22\,\%$; see section 3. If we now use $E_m = 2.1\,\mathrm{GPa}$ for the modulus of the matrix, the rule-of-mixtures gives a prediction of $E_c \approx 22.8\,\mathrm{GPa}$ for the modulus of the composite. This value is in the range of our measured modulus $E_{33}^{\mathrm{UCIIa}} = 23.3\,\mathrm{GPa}$; compare Table 9.

Now we consider the system CIIa. With a volume fraction of $3.5\,\%$, the rule-of-mixtures predicts a modulus of $E_c \approx 16\,\mathrm{GPa}$, which is substantially larger than the results of our measurements; see Table 9. But we can expect this discrepancy, because the carbon nanotube is with a length of about 7 nm too short in comparison to the size of the unit cell of the composite. In particular, a typical single-walled carbon nanotube is approximately 250 times longer. To overcome this finite size effect, we follow Liu and Chen [35, 36] and employ an extended rule-of-mixtures

$$E_c^{\mathrm{ex}} = \left(\frac{1}{E_m} \frac{(L - L_c)}{L} + \frac{1}{E_c} \frac{L_c}{L} \frac{A}{A_c} \right)^{-1} , \tag{89}$$

which also takes the distribution of the fiber into account; see Figure 28. This extended rule-of-mixtures gives substantially better predictions for the modulus of the composite systems CIa, $CIIa$, $CIIb$, and $CIIc$ as can been seen in Table 11. The remaining values that we used to employ equation (89) are also given in Table 11. Note

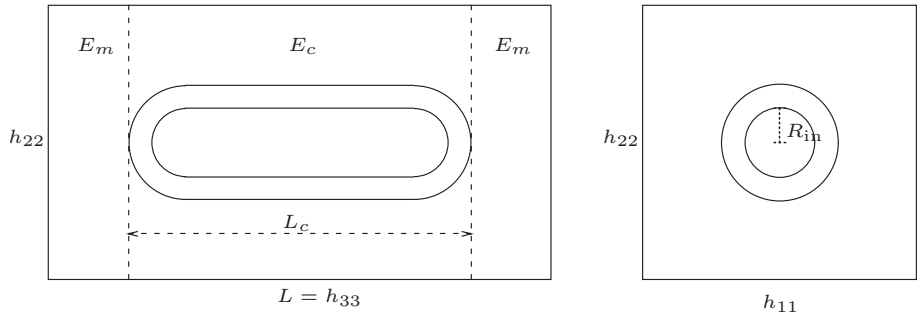


Figure 28: Schematic diagram of a unit cell. Here, we define the areas $A = h_{11}h_{22}$ and $A_c = A - \pi R_{\text{in}}^2$. Furthermore, we assume $R_{\text{in}} = 5\,\text{Å}$.

Table 11: The predicted moduli for the rule-of-mixtures given in equation (88) and for the extended rule-of-mixtures given in equation (89). For the modulus of the nanofiber E_f we used a value of 398.77 GPa, for the length of the nanofiber L_c a value of 67.5 Å, for the modulus of the matrix E_m we used a value of 2.1 GPa in the case of model I and a value of 2.9 GPa in the case of model II.

System	E_c[GPa]	Ω_f	E_c^{ex}[GPa]	L[Å]	A[Å^2]	A_c[Å^2]
CIIa	16.06	0.0352	5.70	96.7	2960.9	2944.3
CIIb	11.66	0.0241	4.21	121.7	2427.2	3421.6
CIIc	9.24	0.0180	3.74	139.0	4027.0	4010.4
MIa	20.97	0.0457	8.23	91.0	2427.2	2410.5

that there is a difference between the Young's modulus of the polyethylene matrix in the case of model I ($E_{33}^{\mathrm{MIa}} \approx 2.1\,\mathrm{GPa}$), and the Young's modulus of the polyethylene matrix in the case of model II ($E_{33}^{\mathrm{MIIa}} \approx 2.9\,\mathrm{GPa}$). Also the Young's modulus of system CIa and of system CIIa show a difference of about one-half gigapascal; see Table 9 and Table 10.

Finally, we consider the systems CIa, F1Ia, F5Ia, and F10Ia with respect to the effect of the cross-links on the Young's modulus for the longitudinal load. Here, our results gave a reinforcement in comparison to system CIa. We obtained a modulus that was 1.04 times higher for system F1Ia, a modulus that was 1.14 times higher for system F5Ia, and a modulus that was 1.17 times higher for system F10Ia than the modulus of system CIa; see Table 10. Note that we used here model I, which allows us to accurately describe the covalent bonds between the matrix and the nanotube. Altogether, we anticipate that a longer and properly functionalized nanotube would lead to improved reinforcement. Note also that especially in the case of functionalized carbon nanotubes, the second-generation REBO potential [188] can be used to model the hydrocarbon system with enhanced accuracy.

Further results from molecular dynamics simulations of the influence of chemical cross-links on the elastic moduli of polymer-carbon nanotube composites are given in [189]. Finally in [190], we studied a composite consisting of boron-nitride nanotubes and silicon-boron-nitride ceramics. There, the dependency of the Young's modulus on the BN-nanotube/$Si_3B_3N_7$-matrix ratio is investigated.

5 Concluding Remarks

The numerical ingredients and the achieved results of a molecular dynamics simulation study for the analysis of elastic properties of a carbon nanotube/polyethylene composite was presented. Here, we used the Parrinello-Rahman technique to apply external stress to generate stress-strain curves. Due to the large amount of particles in the composite, an efficient implementation using a modified linked cell approach is necessary. Furthermore, a parallelization of the code is needed that is based on domain decomposition to exploit parallel computing systems with distributed memory. As model problems, we considered a periodically replicated carbon nanotube that was embedded into a polyethylene matrix and a short-capped nanotube that was embedded in polyethylene matrices of different sizes. To study the reinforcement of the matrix, load was applied and the resulting Young's modulus was calculated. These Young's moduli were compared to two different rules-of-mixtures. The simple rule, which only takes the volume fraction of the fiber into account, holds for the periodically replicated uncapped nanotube but fails for the composites with the short capped nanotube. The extended rule, which also takes the orientation of the tube into account, substantially improves the predictions for the case of the fully embedded short nanotube. Here, we used Brenner's potential for the nanotube and an united-atom potential for the polyethylene matrix. To study the effect of cross-links between the polyethylene matrix and the nanotube, we employed Brenner's reactive bond-order potential for both the matrix and the nanotube. We found that the reinforcement of the matrix increases with the amount of cross-links.

The simulation results suggest the possibility to use nanotubes to reinforce appropriate matrix material. They furthermore indicate that long functionalized nanotubes

should be used. For a fixed tensile loading direction, the nanotubes should be aligned parallel to the loading direction. For general kinds of loading directions, very long nanotubes in random orientation will most likely produce the best results. This, however, is the subject of future studies.

Acknowledgments

This work is based in part on the results of a collaboration with S. J. V. Frankland, A. Caglar, and D. W. Brenner. It was supported in part by grants from the *Schwerpunktprogramm 1165*, the *Sonderforschungsbereich 408*, and the *Sonderforschungsbereich 611* of the *Deutsche Forschungsgemeinschaft*. We thank our colleagues M. Arndt, L. Jager, and R. Wildenhues, who were involved in the development of our molecular dynamics software package.

References

[1] H. W. Kroto, J. R. Heath, S. C. O'Brien, R. F. Curl, and R. E. Smalley. C_{60} Buckminsterfullerene. *Nature*, 318:162–163, 1985.

[2] S. Iijima. Helical microtubules of graphitic carbon. *Nature*, 354:56–58, 1991.

[3] R. E. Smalley and D. T. Colbert. Self assembly of fullerene tubes and balls. talk at: Robert A. Welch Foundation, October 1995.

[4] D. Qian, G. J Wagner, W. K. Liu, M.-F. Yu, and R. S. Ruoff. Mechanics of carbon nanotubes. *Appl. Mech. Rev.*, 55:495, 2002.

[5] R. S. Ruoff, D. Qian, and W. K. Liu. Mechanical properties of carbon nanotubes: Theoretical predictions and experimental measurements. *C. R. Physique*, 4:993–1008, 2003.

[6] M. M. J. Treacy, T. W. Ebbesen, and J. M. Gibson. Exceptionally high Young's modulus observed for individual carbon nanotubes. *Nature*, 381:678–680, 1996.

[7] B. I. Yakobson and R. E. Smalley. Fullerene nanotubes: $C_{1,000,000}$ and beyond. *American Scientist*, 85(4):324–337, 1997.

[8] P. M. Ajayan, L. S. Schadler, S. C. Giannaris, and A. Rubio. Single-walled carbon nanotube-polymer composites: Strength and weakness. *Adv. Materials*, 12:750–753, 2000.

[9] C. Bower, R. Rosen, L. Jin, J. Han, and O. Zhou. Deformation of carbon nanotubes in nanotube-polymer composites. *Appl. Phys. Lett.*, 74:3317–3319, 1999.

[10] M. Lamy de la Chapelle, C. Stephan, T. P. Nguyen, S. Lefrant, C. Journet, P. Bernier, E. Munoz, A. Benito, W. K. Maser, M. T. Martinez, G. D. de la Fuente, T. Guillard, G. Flamant, L. Alvarez, and D. Laplaze. Raman characterization of singlewalled carbon nanotubes and PMMA-nanotubes composites. *Synthetic Metals*, 103:2510–2512, 1999.

[11] X. Gong, J. Liu, S. Baskaran, R. D. Voise, and J. S. Young. Surfactant-assisted processing of carbon nanotube/polymer composite. *Chem. Mater.*, 12:1049–1052, 2000.

[12] R. Haggemueller, H. H. Gommas, A.G. Rinzler, J. E. Fischer, and K. I. Winey. Aligned single-wall carbon nanotubes in composites by melt processing methods. *Chem. Phys. Lett.*, 330:219–225, 2000.

[13] Z. Jia, Z. Wang, C. Xu, J. Liang, B. Wei, D. Wu, and S. Zhu. Study on poly(methylmethacrylate)/carbon nanotube composites. *Mat. Sci. Eng. A*, 271:395–400, 1999.

[14] L. Jin, C. Bower, and O. Zhou. Alignment of carbon nanotubes in a polymer matrix by mechanical stretching. *Appl. Phys. Lett.*, 73:1197–1199, 1998.

[15] D. Qian, E. C. Dickey, R. Andrews, and T. Rantell. Load transfer and deformation mechanisms in carbon nanotube-polystyrene composites. *Appl. Phys. Lett.*, 76:2868–2870, 2000.

[16] M. S. P. Schaffer and A. H. Windle. Fabrication and characterization of carbon nanotube/poly(vinyl alcohol) composites. *Adv. Materials*, 11:937–941, 1999.

[17] L. S. Schadler, S. C. Giannaris, and P. M. Ajayan. Load transfer in carbon nanotube epoxy composites. *Appl. Phys. Lett.*, 73:3842–3844, 1998.

[18] C. Velasco-Santos, A. L. Martínez-Hernández, F. T. Fisher, R. S. Ruoff, and V. M. Castaño. Improvement of thermal and mechanical properties of carbon nanotube composites through chemical functionalization. *Chem. Mater.*, 15(23):4470–4475, 2003.

[19] D. H. Robertson, D. W. Brenner, and J. W. Mintmire. Energetics of nanoscale graphitic tubules. *Phys. Rev. B*, 45(21):12592–12595, 1992.

[20] J.-C. Charlier, A. De Vita, X. Blase, and R. Car. Microscopic growth mechanisms for carbon nanotubes. *Science*, 275:647–649, 1997.

[21] M. B. Nardelli, B. I. Yakobson, and J. Bernholc. Mechanism of strain release in carbon nanotubes. *Phys. Rev. B*, 57(8):R4277–R4280, 1998.

[22] S. Reich, C. Thomsen, and P. Ordejón. Elastic properties of carbon nanotubes under hydrostatic pressure. *Phys. Rev. B*, 65:153407, 2002.

[23] J. Hamaekers. Ebene-Wellen basiertes, adaptives und paralleles Verfahren für die Dichtefunktionaltheorie. Diplomarbeit, Institut für Angewandte Mathematik, Universität Bonn, Germany, 2002.

[24] E. Hernández, C. Goze, P. Bernier, and A. Rubio. Elastic properties of C and $B_xC_yN_z$ composite nanotubes. *Phys. Rev. Lett.*, 80(20):4502–4505, 1998.

[25] B. I. Yakobson, C. J. Brabec, and J. Bernholc. Nanomechanics of carbon tubes: Instabilities beyond linear response. *Phys. Rev. Lett.*, 76(14):2511–2514, 1996.

[26] C. F. Cornwell and L. T. Wille. Elastic properties of single-walled carbon nanotubes in compression. *Solid State Commun.*, 101:555–558, 1997.

[27] J. P. Lu. Elastic properties of carbon nanotubes and nanoropes. *Phys. Rev. Lett.*, 79(7):1297–1300, 1997.

[28] C. F. Cornwell and L. T. Wille. Simulations of the elastic response of single-walled carbon nanotubes. *Computational Materials Science*, 10:42–45, 1998.

[29] A. Caglar and M. Griebel. On the numerical simulation of fullerene nanotubes: $C_{100.000.000}$ and beyond! In R. Esser, P. Grassberger, J. Grotendorst, and M. Lewerenz, editors, *Molecular Dynamics on Parallel Computers, NIC, Jülich 8-10 February 1999*. World Scientific, 2000.

[30] D. Qian, W. K. Liu, and R. S. Ruoff. Mechanics of C_{60} in nanotubes. *J. Phys. Chem. B*, 105:10753–10758, 2001.

[31] T. Belytschko, S. P. Xiao, C. Schatz, and R. S. Ruoff. Atomistic simulations of nanotube fracture. *Phys. Rev. B*, 65:235430, 2002.

[32] B. Ni, S. B. Sinnott, P. T. Mikulski, and J. A. Harrison. Compression of carbon nanotubes filled with C_{60}, CH_4, or Ne: Predictions from molecular dynamics simulations. *Phys. Rev. Lett.*, 88(20):205505, 2002.

[33] D. Qian, W. K. Liu, S. Subramoney, and R. S. Ruoff. Effect of interlayer potential on mechanical deformation of multiwalled carbon nanotubes. *Journal of Nanoscience and Nanotechnology*, 3:185–191, 2003.

[34] W. K Liu D. Qian and R. S. Ruoff. Load transfer mechanism in carbon nanotube ropes. *Composites Science and Technology*, 63(11):1561–1569, 2003.

[35] Y. J. Liu and X. L. Chen. Evaluations of the effective material properties of carbon nanotube-based composites using a nanoscale representative volume element. *Mechanics of Materials*, 35:69–81, 2003.

[36] X.L. Chen and Y. J. Liu. Square representative volume elements for evaluating the effective material properties of carbon nanotube-based composites. *Computational Materials Science*, 29(1):1–11, 2004.

[37] S. J. V. Frankland, A. Caglar, D. W. Brenner, and M. Griebel. Reinforcement mechanisms in polymer nanotube composites: Simulated non-bonded and cross-linked systems. *Mat. Res. Soc. Symp. Proc.*, pages A.1417.1–5, 2000.

[38] S. J. V. Frankland, A. Caglar, D. W. Brenner, and M. Griebel. Molecular simulation of the influence of chemical cross-links on the shear strength of carbon nanotube-polymer interfaces. *J. Phys. Chem. B*, 106:3046–3048, 2002.

[39] S. J. V. Frankland, V. M. Harik, G. M. Odegard, D. W. Brenner, and T. S. Gates. The stress-strain behavior of polymer-nanotube composites from molecular dynamics simulations. *Composites Science and Technology*, 63(11):1655–1661, 2003.

[40] D. N. Theodorou and U. W. Suter. Atomistic modeling of mechanical properties of polymeric glasses. *Macromolecules*, 19:139–154, 1986.

[41] K. W. Wojciechowski, K. V. Tretiakov, A. C. Brańka, and M. Kowalik. Elastic properties of two-dimensional hard disks in the close-packing limit. *J. Chem. Phys.*, 119(2):939–946, 2003.

[42] M. Parrinello and R. Rahman. Crystal structure and pair potentials: A molecular-dynamics study. *Phys. Rev. Lett.*, 45(14):1196–1199, 1980.

[43] M. Parrinello and A. Rahman. Strain fluctuations and elastic constants. *J. Chem. Phys.*, 76(5):2662–2666, 1982.

[44] J. R. Ray and A. Rahman. Statistical ensembles and molecular dynamics studies of anisotropic solids. *J. Chem. Phys.*, 80:4423–4428, 1984.

[45] J. R. Ray and A. Rahman. Statistical ensembles and molecular dynamics studies of anisotropic solids. II. *J. Chem. Phys.*, 82:4243–4247, 1985.

[46] J. R. Ray. Elastic constants and statistical esembles in molecular dynamics. *Comp. Phys. Rep.*, 8:109–152, 1988.

[47] A. A. Gusev, M. M. Zehnder, and U. W. Suter. Fluctuation formula for elastic constants. *Phys. Rev. B*, 54(1):1–4, 1996.

[48] M. Karimi, G. Stapay, T. Kaplan, and M. Mostoller. Temperature dependence of the elastic constants of Ni: Reliability of EAM in predicting thermal properties. *Modelling Simul. Mater. Sci. Eng.*, 5:337–346, 1997.

[49] Z. Zhou. Fluctuations and thermodynamics properties of the constant shear strain ensemble. *J. Chem. Phys.*, 114(20):8769–8774, 2001.

[50] M. Parrinello and A. Rahman. Polymorphic transitions in single crystals: A new molecular dynamics method. *J. Appl. Phys.*, 52(12):7182–7190, 1981.

[51] H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola, and J. R. Haak. Molecular dynamics with coupling to an external bath. *J. Chem. Phys.*, 81:3684–3690, 1984.

[52] D. Brown and J. H. R. Clarke. Molecular dynamics simulation of an amorphous polymer under tension. 1. Phenomenology. *Macromolecules*, 24:2075–2082, 1991.

[53] S. Blonski, W. Brostow, and J. Kubát. Molecular-dynamics simulations of stress relaxation in metals and polymers. *Phys. Rev. B*, 49(10):6494–6500, 1994.

[54] S. Nosé and M. L. Klein. Constant pressure molecular dynamics for molecular systems. *J. Mol. Phys.*, 50:1055–1076, 1983.

[55] S. Nosé. A unified formulation of the constant temperature molecular dynamics methods. *J. Chem. Phys.*, 81(1):511–519, 1984.

[56] S. Nosé. A molecular dynamics method for simulations in the canonical ensemble. *Mol. Phys.*, 52(2):255–268, 1984.

[57] W. G. Hoover. Canonical dynamics: Equilibrium phase-space distributions. *Phys. Rev. A*, 31(3):1695–1697, 1985.

[58] S. Nosé. Constant temperature molecular dynamics methods. *Prog. Theor. Phys. Supp.*, 103:1–46, 1991.

[59] D. W. Brenner. Empirical potential for hydrocarbons for use in simulating the chemical vapor deposition of diamond films. *Phys. Rev. B*, 42(15):9458–9471, 1990.

[60] D. W. Brenner. Erratum: Empirical potential for hydrocarbons for use in simulating the chemical vapor deposition of diamond films. *Phys. Rev. B*, 46(3):1948, 1990.

[61] S. H. Lee, H. Lee, H. Pak, and J. C. Rasaiah. Molecular dynamics simulation of liquid alkanes. I. Thermodynamics and structures of normal alkanes: n-butane to n-heptadecane. *Bull. Korean Chem. Soc.*, 17:735–744, 1996.

[62] M. P. Allen and D. J. Tildesley. *Computer Simulation of Liquids*. Oxford University Press, 1987.

[63] V. Arnold. *Mathematical methods of classical mechanics*. Springer, New York, 1978.

[64] J. L. Lebowitz, J. K. Percus, and L. Verlet. Ensemble dependence of fluctuations with application to machine computations. *Phys. Rev.*, 153(1):250–254, 1967.

[65] D. Frenkel and B. Smit. *Understanding molecular simulation: From algorithms to applictions*. Academic Press, New York, 1996.

[66] G. R. Fowles and G. L. Cassiday. *Analytical Mechanics*. Saunders, New York, 5th edition, 1993.

[67] T. L. Chow. *Classical Mechanics*. Wiley, New York, 1995.

[68] D. T. Greenwood. *Classical Dynamics*. Dover Pubns, 1997.

[69] J. V. José and E. J. Saletan. *Classical Dynamics : A Contemporary Approach*. Cambridge University Press, 1998.

[70] M. E. Tuckerman, Y. Liu, G. Ciccotti, and G. J. Martyna. Non-Hamiltonian molecular dynamics: Generalizing Hamiltonian phase space principles to non-Hamiltonian systems. *J. Chem. Phys.*, 115:1678–1702, 2001.

[71] E. Hernández. Metric-tensor flexible-cell algorithm for isothermal-isobaric molecular dynamics simulations. *J. Chem. Phys.*, 115(22):10282–10290, 2001.

[72] S. D. Bond, B. J. Leimkuhler, and B. B. Laird. The Nosé-Poincaré method for constant temperature molecular dynamics. *J. Comp. Phys.*, 115:114–134, 1999.

[73] C. L. Cleveland. New equations of motion for molecular dynamics systems that change shape. *J. Chem. Phys.*, 89(8):4987–4993, 1988.

[74] R. M. Wentzcovitch. Invariant molecular-dynamics approach to structural phase transitions. *Phys. Rev. B*, 44(5):2358–2361, 1991.

[75] M. Ferrario and J. P. Ryckaert. Constant pressure-constant temperature molecular dynamics for rigid and partially rigid molecular systems. *Mol. Phys.*, 54(3):587–603, 1985.

[76] J. V. Lill and J. Q. Broughton. Nonlinear molecular dynamics and Monte Carlo algorithms. *Phys. Rev. B*, 46:12068–12071, 1992.

[77] G. J. Martyna, D. J. Tobias, and M. L. Klein. Constant pressure molecular dynamics algorithms. *J. Chem. Phys.*, 101(5):4177–4189, 1994.

[78] I. Souza and J. L. Martins. Metric tensor as the dynamical variable for variable-cell-shape molecular dynamics. *Phys. Rev. B*, 55(14):8733–8742, 1997.

[79] M. E. Tuckerman and G. J. Martyna. Understanding modern molecular dynamics: Techniques and applications. *J. Phys. Chem. B*, 104:159–178, 2000.

[80] B. Brooks, R. Bruccoleri, B. Olafson, D. States, S. Swaminathan, and M. Karplus. CHARMM: A program for macromolecular energy minimization, and dynamics calculations. *J. Comput. Chem.*, 4:187–217, 1983.

[81] D. Pearlman, D. Case, J. Caldwell, W. Ross, T. Cheatham, S.Debolt, D. Ferguson, S. Seibel, and P. Kollman. AMBER, a computer program fr applying molecular mechanics, normal mode analysis, molecular dynamics and free energy calculations to elucidate the structures and energies of molecules. *Comp. Phys. Comm.*, 1995.

[82] W. Smith. Calculating the pressure. *CCP5 Info. Quart.*, 39:14–21, 1993.

[83] J. M. Sanz-Serna and M. P. Calvo. *Numerical Hamiltonian Problems*. Chapman & Hall, London, 1994.

[84] W. Swope, H. Andersen, P. Berens, and K. Wilson. A computer simulation method for calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *J. Chem. Phys.*, 76:637–649, 1982.

[85] C. Störmer. Sur les trajectoires des corpuscles életrisés dans l'espace sous l'action du magnetisme terrestre avec application aux aurores boréals. *Arch. Sci. Phys. Nat.*, 24:221–247, 1907.

[86] L. Verlet. Computer "Experiments" on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules. *Phys. Rev.*, 159:98–103, 1967.

[87] R. Hockney. The potential calculation and some applications. *Methods Comp. Phys.*, 9:136–211, 1970.

[88] D. Beeman. Some multistep methods for use in molecular dynamics calculations. *J. Comp. Phys.*, 20:130–139, 1976.

[89] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes in C: The art of scientific computing*. Cambridge University Press, second edition, 1992.

[90] C. W. Gear. *Numerical initial value problems in ordinary differential equations*. Prentice-Hall, Englewood Cliffs, NJ, 1971.

[91] K. Refson. Molecular dynamics simulation of solid n-butane. *Physica B*, 131:256–266, 1985.

[92] M. Griebel, A. Caglar, S. Knapek, and G. Zumbusch. *Numerische Simulation in der Moleküldynamik. Numerik, Algorithmen, Parallelisierung, Anwendungen*. Springer, Berlin, Heidelberg, 2003.

[93] E. Hairer, C. Lubich, and G. Wanner. *Geometric numerical integration. Structure-preserving algorithms for ordinary differential equations*, volume 31 of *Series in Computational Mathematics*. Springer, Berlin, 2nd edition, 2002.

[94] E. Hairer and G. Wanner. Order conditions for general two-step Runge-Kutta methods. *SIAM Journal on Numerical Analysis*, 34(6):2087–2089, 1997.

[95] E. Hairer and C. Lubich. The life-span of backward error analysis for numerical integrators. *Numerische Mathematik*, 76(4):441–462, 1997.

[96] R. Ruth. A canonical integration technique. *IEEE Trans. Nucl. Sci.*, NS-30:2669–2671, 1983.

[97] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I, Nonstiff Problems*, volume 8 of *Springer Series in Computational Mathematics*. Springer, Berlin, 2nd edition, 1991.

[98] H. Trotter. On the product of semi-groups of operators. *Proc. Am. Math. Soc.*, 10:545–551, 1959.

[99] G. Strang. On the construction and comparison of difference schemes. *SIAM J. Numr. Anal.*, 5:506–517, 1968.

[100] E. Forest and R. Ruth. Fourth order symplectic integration. *Physica D*, 43:105–117, 1990.

[101] R. I. McLachlan and G. R. W. Quispel. Splitting methods. *Acta Numerica*, 11:341–434, 2002.

[102] H. Grubmüller. Dynamiksimulationen sehr großer Makromoleküle auf einem Parallelrechner. Diplomarbeit, TU München, 1989.

[103] M. Tuckerman, G. J. Martyna, and B. Berne. Molecular dynamics algorithm for multiple time scales: System with long range forces. *J. Chem. Phys.*, 93:1287–1291, 1990.

[104] H. Grubmüller, H. Heller, A. Windemuth, and K. Schulten. Generalized Verlet algorithm for efficient molecular dynamics simulation with long range interaction. *Mol. Sim.*, 6:121–142, 1991.

[105] M. Tuckerman, B. Berne, and G. J. Martyna. Molecular dynamics algorithm for multiple time scales: System with long range forces. *J. Chem. Phys.*, 94:6811–6815, 1991.

[106] M. Tuckerman, B. Berne, and G. J. Martyna. Reversible multiple time scale molecular dynamics. *J. Chem. Phys.*, 97:1990–2001, 1991.

[107] P. Procacci, M. Marchi, and G. J. Martyna. Electrostatic calculations and multiple time scales in molecular dynamics simulation of flexible molecular systems. *J. Chem. Phys.*, 108:8799–8803, 1998.

[108] M. Marchi and P. Procacci. Coordinates scaling and multiple time step algorithms for simulation of solvated proteins in the NPT ensemble. *J. Chem. Phys.*, 109:5194–5202, 1998.

[109] J. Ryckaert, G. Ciccotti, and H. Berendsen. Numerical integration of the cartesian equation of motion of a system with constraints: Molecular dynamics of N-alkanes. *J. Comput. Phys.*, 23:327–341, 1977.

[110] H. Andersen. Rattle: A 'velocity' version of the Shake alogrithm for molecular dynamics calculations. *J. Comput. Phys.*, 52:24–34, 1983.

[111] J. E. Jr. Dennis and R. B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations.* Prentice-Hall, Englewood Cliffs, NJ, 1983.

[112] J. Nocedal. Theory of algorithms for unconstrained optimization. *Acta Numerica*, 1:199–242, 1992.

[113] E. Polak. *Optimization: Algorithms and Consisten Approximations.* Springer-Verlag, New York, 1997.

[114] R. Fletcher. *Practical Methods of Optimization.* John Wiley & Sons, New York, 2nd edition, 2000.

[115] R. Fletcher and C. M. Reeves. Function minimization by conjugate gradients. *Computer Journal*, 7:148–154, 1964.

[116] E. Polak and G. Ribiére. Note sur la convergence de methodes de directions conjugées. *Revue Française d'Informatique et de Recherche Operationelle*, 16:35–43, 1969.

[117] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Res. NBS*, 49:409–436, 1952.

[118] E. M. L. Beale. *A derivation of conjugate gradient method.* Numerical Methods for Nonlinear Optimization. Academic Press, London, 1972.

[119] J. M. D. Powell. Restart procedures for the conjugate gradient method. *Math. Programming*, 12:241–254, 1977.

[120] M. J. D. Powell. Nonconvex minimization calculations and the conjugate gradient method. In G. A. Watson, editor, *Numerical analysis (Dundee, 1983)*, volume 1066 of *Lecture Notes in Math.*, pages 122–141. Springer, Berlin, 1984.

[121] M. Al-Baali. Descent property and global convergence of the Fletcher-Reeves method with inexact line search. *IMA Journal of Numerical Analysis*, 5:121–124, 1985.

[122] J. C. Gilbert and J. Nocedal. Global convergence properties of conjugate gradient methods for optimization. *SIAM J. Optimization*, 2:21–42, 1992.

[123] L. Armijo. Minimization of functions having Lipschitz continous first partial derivatives. *Pacific J. Mathematics*, 16:1–3, 1966.

[124] P. Wolfe. Convergence conditions for ascent methods. *SIAM Rev.*, 11:226–235, 1969.

[125] P. Wolfe. Convergence conditions for ascent methods. II: Some corrections. *SIAM Rev.*, 13:185–188, 1969.

[126] H. P. Crowder and P. Wolfe. Linear convergence of conjugate gradiant method. *IBM J. Res. and Dev.*, 16:431–433, 1972.

[127] W. C. Davidon. Variable metric methods for minimization. Technical Report ANL-5990, Argonne National Labs, Argonne, IL, 1959.

[128] R. Fletcher and M. J. D. Powell. A rapidly convergent desent method for minimization. *Computer J.*, 3:163–168, 1963.

[129] C. G. Broyden. Quasi-Newton methods and their application to function minimization. *Math. of Computation*, 21:368–381, 1967.

[130] R. Fletcher. A new approach for variable metric algorithms. *Computer J.*, 13:317–322, 1970.

[131] D. Goldfarb. A family of variable metric methods derived by variational means. *Maths. of Computation*, 24:23–26, 1970.

[132] D. F. Shanno. Conditioning of quasi-Newton methods for function minimization. *Maths. of Computation*, 24:647–656, 1970.

[133] J. L. Morales and J. Nocedal. An enriched algorithm for large-scale nonlinear optimization. *Computational Optimization and Applications (COAP)*, 21(2):143–154, 2002.

[134] R. Car and M. Parrinello. Unified approach for molecular dynamics and density-functional theory. *Phys. Rev. Lett.*, 55(22):2471–2474, 1985.

[135] D. Marx and J. Hutter. Ab initio molecular dynamis: Theory and implemantation. In J. Grotendorst, editor, *Modern Methods and Algorithms of Quantum Chemistry*, volume 1 of *NIC Series*, page 301, Jülich, 2000. John von Neumann Institute for Computing, NIC-Directors.

[136] G. C. Abell. Empirical chemical pseudopotential theory of molecular and metallic bonding. *Phys. Rev. B*, 31(10):6184–6197, 1985.

[137] J. Tersoff. New empirical model for the structural properties of silicon. *Phys. Rev. Lett.*, 56(6):632–635, 1986.

[138] J. Tersoff. New empirical approach for structure and energy of covalent systems. *Phys. Rev. B*, 37(12):6991–7000, 1988.

[139] J. Tersoff. Empirical interatomic potential for carbon, with applications to amorphous-carbon. *Phys. Rev. Lett.*, 61(25):2879–2882, 1988.

[140] J. Tersoff. Modeling solid-state chemistry: Interatomic potentials for multicomponent systems. *Phys. Rev. B*, 39(8):5566–5568, 1989.

[141] L. Pauling, L. O. Brockway, and J. Y. Beach. The dependence of interatomic distance on single bond-double bond resonance. *J. Am. Chem. Soc.*, 57:2705–2709, 1935.

[142] L. Pauling. *The Nature of the Chemical Bond*. Cornell University Press, Ithaca, 3rd edition, 1960.

[143] J. A. Harrison, C. T. White, R. J. Colton, and D. W. Brenner. Nanoscale investigation of indentation, adhesion and fracture of diamond (111) surfaces. *Surface Sci.*, 271(1-2):57–67, 1992.

[144] J. A. Harrison, C. T. White, R. J. Colton, and D. W. Brenner. Molecular-dynamics simulations of atomic-scale friction of diamond surfaces. *Phys. Rev. B*, 46(15):9700–9708, 1992.

[145] J. A. Harrison, R. J. Colton, C. T. White, and D. W. Brenner. Effect of atomic-scale surface-roughness on friction – a molecular-dynamics study of diamond surfaces. *Wear*, 168(1-2):127–133, 1993.

[146] J. A. Harrison, C. T. White, R. J. Colton, and D. W. Brenner. Effects of chemically-bound, flexible hydrocarbon specias on the frictional-properties of diamond surfaces. *J. Phys. Chem.*, 97(25):6573–6576, 1993.

[147] J. A. Harrison, C. T. White, R. J. Colton, and D. W. Brenner. Atomistic simulations of friction at sliding diamond interfaces. *MRS Bulletin*, 18(5):50–53, 1993.

[148] J. A. Harrison and D. W. Brenner. Simulated tribochemistry – an atomic-scale view of wear of diamond. *J. Am. Chem. Soc.*, 116(23):10399–10402, 1994.

[149] J. A. Harrison, C. T. White, R. J. Colton, and D. W. Brenner. Investigation of atomic-scale friction and energy-dissipation in diamond using molecular-dynamics. *Thin Solid Films*, 260(2):205–211, 1995.

[150] R. S. Taylor and B. J. Garrison. Hydrogen abstraction reactions in the kiloelectronvolt particle bombardment of organic films. *J. Am. Chem. Soc.*, 116(10):4465–4466, 1994.

[151] J. A. Harrison, D. W. Brenner, C. T. White, and R. J. Colton. Atomistic mechanisms of adhesion and compression of diamond surfaces. *Thin Solid Films*, 206(1-2):213–219, 1991.

[152] S. B. Sinnott, R. J. Colton, C. T. White, and D. W. Brenner. Surface patterning by atomically-controlled chemical forces: Molecular dynamics simulations. *Surf. Sci.*, 316(1-2):L1055–L1060, 1994.

[153] J. Peploski, D. L. Thompson, and L. M. Raff. Molecular dynamics studies of elementary surface reactions of acetylene and ethynyl radical in low-pressure diamond-film formation. *J. Phys. Chem*, 96(21):8538–8544, 1992.

[154] X. Y. Chang D. L. Thompson and L. M. Raff. Hydrogen-atom migration on a diamond (111) surface. *J. Chem. Phys.*, 100(2):1765–1766, 1993.

[155] D. R. Alfonso and S. E. Ulloa. Molecular-dynamics simulations of methyl-radical deposition on diamond (100) surfaces. *Phys. Rev. B*, 48:12235–12239, 1993.

[156] E. R. Williams, G. C. Jones, L. Fang, R. N. Zare B. J. Garrison, and D. W. Brenner. Ion pickup of large, surface-adsorbed molecules: a demonstration of the Eley-Rideal mechanism. *J. Am. Chem. Soc.*, 114(9):3207–3210, 1992.

[157] D. W. Brenner, J. A. Harrison, C. T. White, and R. J. Colton. Molecular dynamics simulations of the nanometer-scale mechanical properties of compressed Buckminsterfullerene. *Thin Solid Films*, 206(1-2):220–223, 1991.

[158] D. H. Robertson D. W. Brenner and C. T. White. On the way to fullerenes: Molecular dynamics study of the curling and closure of graphitic ribbons. *J. Phys. Chem.*, 96(15):6133–6135, 1992.

[159] D. H. Robertson, D. W. Brenner, and C. T. White. Temperature-dependent fusion of colliding C60 fullerenes from molecular dynamics simulations. *J. Phys. Chem.*, 99(43):15721–15724, 1995.

[160] A. Maiti, C. J. Brabec, C. M. Roland, and J. Bernholc. Growth energetics of carbon nanotubes. *Phys. Rev. Lett.*, 73:2468–2471, 1994.

[161] A. Maiti, C. J. Brabec, C. M. Roland, and J. Bernholc. Theory of carbon nanotube growth. *Phys. Rev. B*, 52:14850–14858, 1995.

[162] C. J. Brabec, A. Maiti, C. M. Roland, and J. Bernholc. Growth of carbon nanotubes: A molecular dynamics study. *Chem. Phys. Lett.*, 236(1-2):150–155, 1995.

[163] Y. Xia, Y. Mu, Y. Ma, S. Li, H. Zhang, C. Tan, and L. Mei. Self-assembly growth of single-wall carbon nanotubes. *Nucl. Instr. and Meth. in Phys. Res. B*, 155:395–402, 1999.

[164] M. B. Nardelli, C. Roland, and J. Bernholc. Theoretical bounds for multiwalled carbon nanotube growth. *Chem. Phys. Lett.*, 296:471–476, 1998.

[165] M. B. Nardelli, C. J. Brabec, A. Maiti, C. M. Roland, and J. Bernholc. Lip-Lip interactions and the growth of multiwalled carbon nanotubes. *Phys. Rev. Lett.*, 80(2):313–316, 1998.

[166] Z. Mao, A. Garg, and S. B. Sinnott. Molecular dynamics simulations of the filling and decorating of carbon nanotubules. *Nanotechnology*, 10:273–277, 1999.

[167] Z. Mao and S. B. Sinnott. A computational study of molecular diffusion and dynamic flow through carbon nanotubes. *J. Phys. Chem. B*, 104:4618–4624, 2000.

[168] S. B. Sinnott, O. A. Shenderova, C. T. White, and D. W. Brenner. Mechanical properties of nanotubule fibers and composites determined from theoretical calculations and simulations. *Carbon*, 36(1-2):1–9, 1998.

[169] M. Griebel and J. Hamaekers. Molecular dynamics simulations of the elastic moduli of polymer-carbon nanotube composites. *Computer Methods in Applied Mechanics and Engineering*, 193(17–20):1773–1788, 2004.

[170] S. J. V. Frankland and V. M. Harik. Analysis of carbon nanotube pull-out from a polymer matrix. *Surf. Sci.*, 525(1-3):L103–L108, 2003.

[171] M. A. Moller, D. J. Tildesley, K. S. Kim, and N. Quirke. Molecular dynamics simulation of a Langmuir-Blodgett film. *J. Chem. Phys.*, 94:8390–8401, 1991.

[172] P. M. Morse. Diatomic molecules according to the wave mechanics. II. Vibrational levels. *Phys. Rev.*, 34:57–64, 1929.

[173] J.-P. Ryckaert and A. Bellemans. Molecular dynamics of liquid alkanes. *Faraday Discussions of the Chemical Society*, 66:95–106, 1978.

[174] S. Chynoweth, U. C. Klomp, and L. E. Scales. Simulation of organic liquids using pseudo-pairwise interatomic forces on a toroidal transputer array. *Comput. Phys. Commun.*, 62:297–306, 1991.

[175] S. Chynoweth, U. C. Klomp, and Y. Michopoulos. Comment on: Rheology of n-alkanes by nonequilibrium molecular dynamics. *J. Chem. Phys.*, 95:3024–3025, 1991.

[176] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles.* McGraw-Hill, New York, 1981.

[177] D. M. Beazley, P. S. Lomdahl, N. Gronbech-Jensen, R. Giles, and P. Tamayo. Parallel algorithms for short-range molecular dynamics. In D. Stauffer, editor, *Annual Reviews of Computational Physics*, volume 3, pages 119–175. World Scientific, 1995.

[178] S. Plimpton. Fast parallel algorithms for short-ranged molecular dynamics. *J. Comp. Phys.*, 117:1–19, 1995.

[179] D. S. Chandrasekharaiah and L. Debnath. *Continuum Mechanics*, chapter 9.2. Academic Press, San Diego, 1994.

[180] T. Belytschko, W. K. Liu, and B. Moran. *Nonlinear Finite Elements for Continua and Structures*, chapter 5. John Wiley & Sons, Chichester, 2000.

[181] http://wissrech.iam.uni-bonn.de/research/projects/parnass2/index.html.

[182] D. Asimov. Geometry of capped nanocylinders. AT&T Labs Technical Memorandum, http://www.reserach.att.com/areas/stat/nano, 1998. (with an Appendix by A. Buja).

[183] D. Brown, P. Mélé, S. Marceau, and N. D. Albérola. A molecular dynamics study of a model nanoparticle embedded in a polymer matrix. *Macromolecules*, 36:1395–1406, 2003.

[184] D. Brown. *The* gmq *user manual*, version 3 edition, 1999. available at http://www.univ-savoie.fr/labos/lmops/brown/gmq.html.

[185] G. M. Odegard, S. J. V. Frankland, and T. S. Gates. The effect of chemical functionalization on mechanical properties of nanotube/polymer composites. In *44th Annual AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, AIAA 2003-1701*, Norfolk, Virginia, April 7-10 2003. American Institute of Aeronautics and Astronautics.

[186] Y. Xia, M. Zhao, Y. Ma, M. Ying, X. Liu, P. Liu, and L. Mei. Tensile strength of single-walled carbon nanotubes with defects under hydrostatic pressure. *Phys. Rev. B*, 65:155415–1, 2002.

[187] G. Dereli and C. Özdoğan. Structural stability and energetics of single-walled carbon nanotubes under uniaxial strain. *Phys. Rev. B*, 67:035416–1, 2003.

[188] D. W. Brenner, O. A. Shenderova, J. A. Harrison, S. J. Stuart, B. Ni, and S. B. Sinnott. A second-generation reactive empirical bond order (REBO) potential energy expression for hydrocarbons. *J. Phys.: Condens. Matter*, 14:783–802, 2002.

[189] M. Griebel, J. Hamaekers, and R. Wildenhues. Molecular dynamics simulations of the influence of chemical cross-links on the elastic moduli of polymer-carbon nanotube composites. In J. Sanchez, editor, *Proceedings 1st Nanoc-Workshop*, LABEIN, Bilbao, Spain, 2005, to appear. also as INS Preprint No. 0503.

[190] M. Griebel and J. Hamaekers. Molecular dynamics simulations of boron-nitride nanotubes embedded in amorphous Si-B-N. submitted to *Computational Materials Science*, 2005. also as INS Preprint No. 0501.