$NpT$ PIMC Simulations of Liquid $H_2/D_2$ Mixtures

eingereicht von
Rudolf Pfeiffer

Wien, im März 2001
Contents

List of Figures v
List of Tables vii
Abstract ix
Zusammenfassung xi
Introduction xiii

I Statistical Mechanics and Thermodynamics 1

1 Quantum Systems 3
  1.1 Hilbert Spaces .................................................. 3
  1.2 Linear Operators on Hilbert Spaces ................................ 5
    1.2.1 General Properties ............................................. 5
    1.2.2 Classification of Linear Operators ............................ 6
    1.2.3 Some Remarks on Linear Operators ............................ 8
  1.3 The Hilbert Spaces $\ell^2$ and $L^2(\mathbb{R}^n)$ ..................... 9
  1.4 The Postulates of Quantum Mechanics ................................ 9
  1.5 Pure and Mixed States .......................................... 10

2 Classical Systems 11
  2.1 Phase Space and Micro-States .................................... 11
  2.2 Classical Statistical Mechanics .................................. 12
    2.2.1 Canonical Ensemble ........................................... 12
    2.2.2 Isothermal-Isobaric Ensemble ................................ 15

3 Path-Integral Formalism 19
  3.1 Mathematical Preliminaries ...................................... 19
  3.2 Canonical Ensemble ............................................. 20
    3.2.1 The Density Operator for a Free Particle .................... 20
    3.2.2 One Particle in One Dimension ................................ 20
    3.2.3 Classical Isomorphism ......................................... 23
    3.2.4 $N$ particles in Three Dimensions ............................ 23
## List of Figures

1.1 Schematic classification of linear operators on a Hilbert space $\mathcal{H}$ ............................ 6

3.1 A particle traveling from $x^{(0)}$ to $x^{(P)}$ along a path $x^{(1)}, x^{(2)}, \ldots, x^{(P-1)}$ ........ 22

3.2 Classical isomorphism for a quantum particle ......................................................... 23

5.1 Naive Monte Carlo integration .......................................................... 33

5.2 Algorithm for the Metropolis method (asymmetric rule) ........................................ 37

6.1 The Lennard-Jones potential .......................................................... 40

6.2 Algorithm for MC simulations in the canonical ensemble ........................................ 41

6.3 Accumulated average and block mean values for the density of $D_2$ at $p = 1.5$ atm and $T = 20.4$ K .......................................................... 43

6.4 Algorithm for MC simulations in the isothermal-isobaric ensemble .................................. 47

8.1 Saturation vapor pressure curves for $H_2$ and $D_2$ ................................................. 55

8.2 Phase diagram for classical LJ $H_2$ and $D_2$ in the $p, T$ plane .................................... 57

8.3 Phase diagram for classical LJ $H_2$ and $D_2$ in the $\rho, T$ plane .................................. 58

8.4 Results for pure $H_2$ and $D_2$ at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ .......... 61

8.5 Fit of the densities of $D_2$ at $p = 1.5$ atm and $T = 20.4$ K with $P = 8, 16, 32$ and $64$ and $N = 108$ .......................................................... 63

8.6 Pair-correlation functions for $H_2$ at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ ..... 64

8.7 Pair-correlation functions for $D_2$ at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ ..... 66

8.8 Pair-correlation functions for $H_2, H_2^*, D_2$ and $D_2^*$ at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ and $P = 64$ .......................................................... 70

8.9 Results for pure $H_2$ and $D_2$ at $p = 1.5$ atm and $T = 20.4$ K with $P = 64$ ......... 71

8.10 Pair-correlation functions for $H_2$ and $D_2$ at $p = 1.5$ atm and $T = 20.4$ K with $N = 256$ and $P = 64$ .......................................................... 72

8.11 Results for pure $H_2$ and $D_2$ at $p = 747.3$ atm and $T = 70.4$ K with $N = 108$ .... 74

8.12 Pair-correlation functions for $H_2$ at $p = 747.3$ atm and $T = 70.4$ K with $N = 108$ 76

8.13 Pair-correlation functions for $D_2$ at $p = 747.3$ atm and $T = 70.4$ K with $N = 108$ 78

8.14 Comparison of the pair-correlation functions of $H_2$ and $D_2$ in the states A and B with $N = 108$ and $P = 64$ .......................................................... 79

9.1 Excess volumes for $H_2/D_2$ mixtures at $p = 1.5$ atm and $T = 20.4$ K .................. 88

9.2 Excess enthalpies for $H_2/D_2$ mixtures at $p = 1.5$ atm and $T = 20.4$ K ............... 89
List of Figures

9.3 Pair-correlation functions for a 50% mixture of H\textsubscript{2} and D\textsubscript{2} at p = 1.5 atm and 
\(T = 20.4\) K with \(N = 108\) and \(P = 64\) ............................... 90
9.4 Pair-correlation functions for a 50% mixture of H\textsubscript{2} and D\textsubscript{2} at p = 747.3 atm and 
\(T = 70.4\) K with \(N = 108\) and \(P = 64\) ............................... 91
A.1 Definition of the Legendre transformation .................................. 98
A.2 Commutative diagrams for the Legendre transformation ................. 101
List of Tables

6.1 Reduced units ......................................................... 45
8.1 Results for pure $H_2$ at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ ............. 60
8.2 Results for pure $D_2$ at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ ............. 60
8.3 Maxima and minima of the pair-correlation functions of $H_2$ at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ ............. 65
8.4 Maxima and minima of the pair-correlation functions of $D_2$ at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ ............. 67
8.5 Results for pure $H_2$, $H_2^*$, $D_2$ and $D_2^*$ at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ 68
8.6 Results for pure $H_2$ at $p = 1.5$ atm and $T = 20.4$ K with $P = 64$ ............. 69
8.7 Results for pure $D_2$ at $p = 1.5$ atm and $T = 20.4$ K with $P = 64$ ............. 69
8.8 Results for pure $H_2$ at $p = 747.3$ atm and $T = 70.4$ K with $N = 108$ ............. 73
8.9 Results for pure $D_2$ at $p = 747.3$ atm and $T = 70.4$ K with $N = 108$ ............. 73
8.10 Maxima and minima of the pair-correlation functions of $H_2$ at $p = 747.3$ atm and $T = 70.4$ K with $N = 108$ ............. 77
8.11 Maxima and minima of the pair-correlation functions of $D_2$ at $p = 747.3$ atm and $T = 70.4$ K with $N = 108$ ............. 78
8.12 Results for pure $H_2$ and $D_2$ at $p = 747.3$ atm and $T = 70.4$ K with $N = 108$ 80
9.1 Results for mixtures of $H_2$ and $D_2^{(*)}$ at $p = 1.5$ atm and $T = 20.4$ K with $P = 4$ 83
9.2 Results for mixtures of $H_2$ and $D_2^{(*)}$ at $p = 1.5$ atm and $T = 20.4$ K with $P = 64$ 84
9.3 Results for 50% mixtures of $H_2$ and $D_2$ at $p = 1.5$ atm and $T = 20.4$ K with $P = 128$ 85
9.4 Properties of ideal mixtures of $H_2$ and $D_2^{(*)}$ at $p = 1.5$ atm and $T = 20.4$ K 86
9.5 Excess properties for mixtures of $H_2$ and $D_2^{(*)}$ at $p = 1.5$ atm and $T = 20.4$ K 87
9.6 Results for mixtures of $H_2$ and $D_2$ at $p = 747.3$ atm and $T = 70.4$ K ............. 92
9.7 Ideal properties for 50% mixtures of $H_2$ and $D_2$ at $p = 747.3$ atm and $T = 70.4$ K 93
9.8 Excess properties for 50% mixtures of $H_2$ and $D_2$ at $p = 747.3$ atm and $T = 70.4$ K 93
Abstract

In this work the Path-Integral Monte Carlo method was used to simulate pure hydrogen and pure deuterium and mixtures of these substances. The simulations were performed in the isothermal-isobaric ($NpT$) ensemble, i.e., pressure and temperature of the system were fixed. In Path-Integral Monte Carlo simulations it is possible to “turn on” quantum mechanics by increasing only one parameter, the Trotter number. A Trotter number of 1 corresponds to a classical simulation, and for larger Trotter numbers the properties of the system converge to the properties of the quantum system.

The pure systems and the mixtures were simulated at $p = 1.5$ atm, $T = 20.4$ K and at $p = 747.3$ atm, $T = 70.4$ K. In the first state the classical (low Trotter number) systems are in the solid state and the more quantum mechanical (higher Trotter number) systems are liquids. In the second state the classical and the quantum mechanical systems are super-critical fluids. For all systems the radii of gyration, the densities, the kinetic and potential energies, the enthalpies and the volumes were calculated. Of special interest were the differences between the simulated enthalpies and volumes of the mixtures and the calculated enthalpies and volumes of ideal mixtures (obtained by linear interpolation between the pure systems). These differences are called excess properties and are important in characterizing mixtures. At $p = 1.5$ atm and $T = 20.4$ K it was possible to compare the calculated excess enthalpies and excess volumes to real measurements by Lambert [53]. The agreement between the simulation results for high Trotter numbers and the measurements is surprisingly good. From the simulations it could be concluded that the excess properties are mainly due to quantum effects caused by the small masses of the hydrogen isotopes.
Abstract
Zusammenfassung


Die Reinsysteme und die Mischungen wurden bei \(p = 1.5\text{ atm}, T = 20.4\text{ K}\) und bei \(p = 747.3\text{ atm}, T = 70.4\text{ K}\) simuliert. Im ersten Zustand sind die klassischen Systeme (kleine Trotter-Zahl) Festkörper und die mehr quantenmechanischen Systeme (höhere Trotter-Zahlen) sind Flüssigkeiten. Im zweiten Zustand sind die klassischen und die quantenmechanischen Systeme überkritische Fluide. Für alle Systeme wurden die Gyrationsradien, die Dichten, die kinetischen und potentiellen Energien, die Enthalpien und die Volumina berechnet. Von besonderem Interesse waren dabei die Differenzen zwischen den simulierten Enthalpien und Volumina und den berechneten Enthalpien und Volumina idealer Mischungen (berechnet durch lineare Interpolation zwischen den Reinsystemen). Diese Differenzen bezeichnet man als Exzeßgrößen und sind wichtig für die Charakterisierung von Mischungen. Für \(p = 1.5\text{ atm}\) und \(T = 20.4\text{ K}\) war es möglich, die berechneten Exzeßenthalpien und die Exzeßvolumina mit Messungen von Lambert [53] zu vergleichen. Die Übereinstimmung zwischen Simulationen für höhere Trotter-Zahlen und Experiment ist erstaunlich gut. Aus den Simulationsergebnissen kann man schließen, daß die Exzeßeigenschaften hauptsächlich durch Quanteneffekte verursacht werden, die von den kleinen Massen der Wasserstoffisotope herrühren.
Zusammenfassung
Introduction

Macroscopic amounts of matter consist of the order of \( \mathcal{O}(N_A) \) “particles”\(^1\) (atoms and/or molecules), where \( N_A = 6.02205(3) \times 10^{23} \text{ mol}^{-1} \) is Avogadro’s number. Without much effort it is only possible to measure macroscopic quantities, such as temperature, pressure, volume, mass,… To measure the microscopic properties of such huge systems is nearly impossible. Nevertheless, the physicist tries to explain the macroscopic observable quantities of matter from the microscopic kinematics and dynamics of the molecules. The means for this is statistical physics where models of the microscopic behavior of a system are condensed in so called partition functions from which the thermodynamical potentials can be derived. The problem is that the partition functions can only be calculated analytically for the most simple models (e.g., the ideal gas). Using more realistic models the partition functions and hence the thermodynamical potentials can only be calculated numerically by computer simulations. Such simulations have the additional advantage that one can perform “measurements” on the system that would be very difficult (e.g., under extreme pressure and temperature conditions) or impossible in reality (e.g., one could observe the change of the macroscopic observables while gradually “turning on” quantum mechanics).

In [53] Lambert published his measurements of the excess enthalpies and excess volumes of hydrogen/deuterium mixtures. The measurements were made with liquids in ortho-para equilibrium at \( p = 1.5 \text{ atm} \) and \( T = 20.4 \text{ K} \) (99.8 \% para-H\(_2\), 98 \% ortho-D\(_2\)) [53]. The aim of the present work was to simulate such hydrogen/deuterium mixtures with the means of the Path-Integral Monte Carlo (PIMC) method in the isothermal-isobaric (\( NpT \)) ensemble. Of special interest was the question, whether the excess properties are mainly caused by quantum or other effects. The simulation results suggest that the excess properties are mainly due to quantum effects.

\(^1\)Depending on the desired accuracy these particles can be considered as the phantom-like mass-points from classical mechanics. More general, they can be subsystems—like atoms or molecules—on their own, as long as their inner structure and interactions can be neglected.
Part I

Statistical Mechanics and Thermodynamics
1 Quantum Systems

Quantum mechanics is a lovely introduction to Hilbert spaces.

– (overheard at a party)

This chapter gives a brief introduction to the mathematics and the postulates of non-relativistic quantum mechanics. It is a mixture of Balian [7, 8], Cohen-Tannoudji et al. [20, 21], Feynman et al. [27], Großmann [35, 36], Landau and Lifschitz [54], Müller [60], Schubert and Weber [78], van Hees [89], Werner [92] and Yosida [95].

The broadest introduction to quantum mechanics with many examples can be found in Cohen-Tannoudji et al. [20, 21]. For a more mathematical introduction see Müller [60]. Though its notation is a little bit old-fashioned, Landau and Lifschitz [54] is a standard reference for non-relativistic quantum mechanics. A modern overview can also be found in Schubert and Weber [78]. Balian [7, 8] is a broad way “From Microphysics to Macrophysics” using quantum mechanics as a tool. The mathematical stage for quantum mechanics is a complex Hilbert space. The appropriate framework for Hilbert spaces is functional analysis. A very readable introduction to it is given in Großmann [35, 36]. Yosida [95] is much broader but less readable. A short overview can also be found in Grosche et al. [34]. Werner [92] gives a modern introduction to functional analysis.

1.1 Hilbert Spaces

In quantum mechanics every physical system is mathematically described by a separable complex Hilbert space $\mathcal{H}$ with elements (or vectors) $|\cdot\rangle$, called kets or state vectors. A Hilbert space is a complete linear vector space with an inner product (or scalar product).

Elements $|\psi_1\rangle, \ldots, |\psi_n\rangle$ in $\mathcal{H}$ are called linearly independent $:\iff \alpha_1|\psi_1\rangle + \cdots + \alpha_n|\psi_n\rangle$ $(\alpha_1, \ldots, \alpha_n \in \mathbb{C})$ holds only in the trivial case $\alpha_1 = \cdots = \alpha_n = 0$; otherwise $|\psi_1\rangle, \ldots, |\psi_n\rangle$ are linearly dependent. $\mathcal{H}$ is finite dimensional and, moreover, $n$-dimensional if $\mathcal{H}$ contains $n$ linearly independent vectors and if any $n+1$ elements of $\mathcal{H}$ are linearly dependent. If $\mathcal{H}$ has arbitrarily many linearly independent vectors then it is infinite dimensional. [3]

Via the inner product $(\cdot, \cdot) : \mathcal{H} \times \mathcal{H} \to \mathbb{C}$ it is possible to define for every fixed ket $|\chi\rangle \in \mathcal{H}$ a bounded linear $^1$ (and hence continuous) functional $\chi : \mathcal{H} \to \mathbb{C}$, with $\chi(|\psi\rangle) := (|\chi\rangle, |\psi\rangle).$ $^2$

The set of these linear functionals on the kets $|\psi\rangle \in \mathcal{H}$ constitutes again a Hilbert space which is called the dual space of $\mathcal{H}$ and which is symbolized by $\mathcal{H}'$. The dual space is conjugate

$^1$The inner product is anti-linear in its first and linear in its second argument (sesqui-linear form). Hence the linearity of the functional.

$^2$After the representation theorem by Friedrich Riesz every bounded linear functional $\chi$ on $\mathcal{H}$ can be represented in the form $\chi(|\psi\rangle) = (|\chi\rangle, |\psi\rangle)$, where the fixed ket $|\chi\rangle$ is uniquely determined by the functional $\chi$; furthermore, $\|\chi\| = |||\chi|||$. 
linear isometric isomorph to the Hilbert space: $\mathcal{H}' \cong^* \mathcal{H}$. The elements of $\mathcal{H}'$ are called bras and are denoted by $\langle \cdot | \cdot \rangle$. Hence the bra $\langle \chi |$ designates the linear functional $\chi$, such that $\langle \chi | \psi \rangle := \chi(|\psi\rangle) = (|\chi\rangle, \psi)$. The bra associated with $\psi$ is $\langle \psi |^* \rangle$, where $\psi^*$ denotes the complex conjugate of $c$. Since $\mathcal{H}'$ is again a Hilbert space, one can construct the bi-dual space $\mathcal{H}'' = (\mathcal{H}')'$ of $\mathcal{H}$.

Two kets $|\phi\rangle$ and $|\psi\rangle$ in $\mathcal{H}$ are orthogonal ($|\phi\rangle \perp |\psi\rangle$) : $\iff \langle \phi | \psi \rangle = 0$.

The inner product induces a norm $\| \cdot \| : \mathcal{H} \rightarrow [0, \infty)$, where $\| |\phi\rangle |\phi\rangle \| := \sqrt{\langle \phi | \phi \rangle}$. Any two vectors $|\phi\rangle, |\psi\rangle \in \mathcal{H}$ fulfill the triangle inequality $\| |\phi\rangle + |\psi\rangle \| \leq \| |\phi\rangle \| + \| |\psi\rangle \|$, and the Cauchy-Schwarz inequality $\| \langle \phi | \psi \rangle \| \leq \| |\phi\rangle \| \| |\psi\rangle \|$, with equality if and only if $|\phi\rangle$ and $|\psi\rangle$ are linearly dependent. Further hold $\| |\phi\rangle + |\psi\rangle \|^2 = \| |\phi\rangle \|^2 + \| |\psi\rangle \|^2$ for all $|\phi\rangle, |\psi\rangle \in \mathcal{H}$ with $|\phi\rangle \perp |\psi\rangle$ (theorem of Pythagoras) and $\| |\phi\rangle + |\psi\rangle \|^2 + \| |\phi\rangle - |\psi\rangle \|^2 = 2(\| |\phi\rangle \|^2 + \| |\psi\rangle \|^2)$ for all $|\phi\rangle, |\psi\rangle \in \mathcal{H}$ (parallelogram rule). The norm in $\mathcal{H}'$ is given by $\| \langle \chi | \| = \sup_{\| |\psi\rangle \| \leq 1} \langle \langle \chi | \psi \rangle \|$. Using the Cauchy-Schwarz inequality one gets $\| \langle \chi | \| = \| |\chi\rangle \|$.\(^3\)

The norm defines a metric or distance function $d(|\phi\rangle, |\psi\rangle) := \| |\phi\rangle - |\psi\rangle \| = \| |\psi\rangle - |\phi\rangle \|$, which in turn generates the strong topology in $\mathcal{H}$: The convergence $\lim_{n \to \infty} \| |\psi_n\rangle - |\psi\rangle \| = 0$ is denoted by $s\lim_{n \to \infty} |\psi_n\rangle = |\psi\rangle$ or simply $|\psi_n\rangle \xrightarrow{s} |\psi\rangle$, and it is said that the sequence $\{ |\psi_n\rangle \}_{n=1}^{\infty}$ converges strongly to $|\psi\rangle$. If for every $|\phi\rangle \in \mathcal{H}$ holds $\lim_{n \to \infty} \langle \langle \phi | \psi_n\rangle \rangle = \langle \langle \phi | \psi \rangle \rangle$ then it is said that the sequence $\{ |\psi_n\rangle \}_{n=1}^{\infty}$ converges weakly to $|\psi\rangle$. This weak convergence is denoted by $w\lim_{n \to \infty} |\psi_n\rangle = |\psi\rangle$ or simply $|\psi_n\rangle \xrightarrow{w} |\psi\rangle$. The strong convergence implies the weak convergence, but not vice versa. A sequence $\{ |\psi_n\rangle \}_{n=1}^{\infty}$ in $\mathcal{H}$ is called a (strong) Cauchy sequence if for every $\epsilon > 0$ exists an $N(\epsilon) \in \mathbb{N}$ such that $\| |\psi_n\rangle - |\psi_m\rangle \| < \epsilon$ for $n, m > N(\epsilon)$. If the sequence $\{ |\psi_n\rangle \}_{n=1}^{\infty}$ is a Cauchy sequence in $\mathbb{C}$ for every $|\phi\rangle \in \mathcal{H}$ then the sequence $\{ |\psi_n\rangle \}_{n=1}^{\infty}$ is called a weak Cauchy sequence. Obviously, every strongly convergent sequence is a Cauchy sequence. Since $\mathcal{H}$ is a complete space every Cauchy sequence in $\mathcal{H}$ is also a strongly convergent sequence. The last two statements are also valid for weak convergence.

A subset $M \subset \mathcal{H}$ is called dense if for every ket $|\psi\rangle \in \mathcal{H}$ and every $\epsilon > 0$ exists a ket $|\phi\rangle \in M$ with $\| |\psi\rangle - |\phi\rangle \| < \epsilon$. I.e., one can approximate every $|\psi\rangle \in \mathcal{H}$ to every desired accuracy by a $|\phi\rangle \in M$. Let $M$ be an arbitrary subset of $\mathcal{H}$. For all Cauchy sequences in $M$ exists a limit in $\mathcal{H}$, because $\mathcal{H}$ is complete. The closure $\overline{M}$ of $M$ is defined as the set of these limits. Clearly, $M \subseteq \overline{M} \subseteq \mathcal{H}$. If $M$ is a dense subset of $\mathcal{H}$ the closure $\overline{M}$ of $M$ is the Hilbert space in the sense of the strong topology: $\overline{M} = \mathcal{H}$. A Hilbert space is called separable if it contains a countable dense subset $M \subseteq \mathcal{H}$.

The separability of $\mathcal{H}$ ensures the existence of at least countable complete orthonormal systems $\{ |k\rangle \}$, that are called Hilbert bases.\(^4\) These complete orthonormal systems satisfy the orthonormality relations $\langle k | k' \rangle = \delta_{k,k'}$, where $\delta_{k,k'}$ is the Kronecker delta, and the closure relation

$$\sum_k |k\rangle \langle k| = \hat{I}, \quad (1.1.1)$$

\(^3\) $\| |\chi\rangle \| = \sup_{\| |\psi\rangle \| \leq 1} \langle |\chi\rangle |\psi\rangle \leq \sup_{\| |\psi\rangle \| \leq 1} \| |\chi\rangle \| \| |\psi\rangle \| \leq \| \| |\chi\rangle \| \|$. On the contrary, using the properties of sup, one gets $\| \| |\chi\rangle \| \| = \sup_{\| |\psi\rangle \| \leq 1} \langle |\chi\rangle |\psi\rangle \geq \frac{1}{\| |\chi\rangle \|} \| |\chi\rangle \| = \| |\chi\rangle \|$.\(^5\) These bases are constructed via Gram-Schmidt orthogonalization of linearly independent vectors of the dense subset $M \subseteq \mathcal{H}$ in $\mathcal{H}$.

\(^5\) Countable orthonormal systems are sometimes called orthonormal sequences.

---

4量子系统
where $\hat{I}$ is the identity operator on $\mathcal{H}$. Two Hilbert bases in one Hilbert space $\mathcal{H}$ have always the same cardinality. The dimension $\dim \mathcal{H}$ of a Hilbert space $\mathcal{H}$ is given by the cardinality of one (and hence all) Hilbert bases in $\mathcal{H}$. Two Hilbert spaces with the same dimension are isomorphic and vice versa.\footnote{Therefore, two abstract Hilbert spaces differ from each other only in their dimension.}

Due to (1.1.1) every ket $|\psi\rangle \in \mathcal{H}$ can be uniquely represented by its coordinates $\langle k|\psi\rangle$

$$|\psi\rangle = \sum_k |k\rangle \langle k|\psi\rangle \quad (1.1.2)$$

and the representation obeys Parseval’s equation $\|\psi\|^2 = \sum_k |\langle k|\psi\rangle|^2$. Since $|k\rangle\langle k|\psi\rangle$ gives the orthogonal projection of $|\psi\rangle$ onto the subspace span$\{|k\rangle\} \subseteq \mathcal{H}$ spanned by $|k\rangle$, the operator $\hat{P}_k := |k\rangle\langle k|$ is called the projection operator or projector on the subspace span$\{|k\rangle\}$.

$\mathbb{R}^n$ and $\mathbb{C}^n$ are some simple examples for Hilbert spaces. More abstract examples—widely used in quantum mechanics and other areas of physics—are the Hilbert spaces $\ell^2$ and $L^2(\mathbb{R}^n)$. They are discussed in section 1.3.

### 1.2 Linear Operators on Hilbert Spaces

#### 1.2.1 General Properties

An operator $\hat{X}$ on $\mathcal{H}$ assigns to every ket $|\psi\rangle \in \mathcal{H}$ another ket $|\psi'\rangle = \hat{X}|\psi\rangle \in \mathcal{H}$. The set of all these kets $|\psi'\rangle$ is called the range ran $\hat{X} \subseteq \mathcal{H}$ of the operator $\hat{X}$. If ran $\hat{X} = \mathcal{H}$ $\hat{X}$ is called surjective. If $\hat{X}|\psi_1\rangle = \hat{X}|\psi_2\rangle$ implies $|\psi_1\rangle = |\psi_2\rangle$ $\hat{X}$ is called injective. If $\hat{X}$ is surjective and injective it is called bijective. If $\hat{X}$ is injective then the inverse operator $\hat{X}^{-1} : \text{ran } \hat{X} \to \text{dom } \hat{X}$ exists.

$\hat{X}$ is a linear operator $\iff \hat{X}(\alpha|\psi_1\rangle + \beta|\psi_2\rangle) = \alpha\hat{X}|\psi_1\rangle + \beta\hat{X}|\psi_2\rangle$ for all $|\psi_1\rangle,|\psi_2\rangle \in \text{dom } \hat{X}$ and all $\alpha,\beta \in \mathbb{C}$. In this linear case dom $\hat{X}$ is necessarily a linear subspace\footnote{In the following, subspaces are meant in the sense vector spaces; they need not be closed.} of $\mathcal{H}$, though it need not be closed (as in finite spaces). The same is true for the range ran $\hat{X}$, though in general it need not be a linear subspace of the same Hilbert space $\mathcal{H}$.

An operator is said to be defined everywhere if dom $\hat{X} = \mathcal{H}$. An operator $\hat{Y}$ with domain $\text{dom } \hat{Y}$ is called an extension of $\hat{X}$ with domain $\text{dom } \hat{X}$ if $\text{dom } \hat{Y} \supset \text{dom } \hat{X}$ and $\hat{Y}|\psi\rangle = \hat{X}|\psi\rangle$ for every $|\psi\rangle \in \text{dom } \hat{X}$. In reverse $\hat{X}$ is called a restriction of $\hat{Y}$.

A linear operator $\hat{X} : \text{dom } \hat{X} \to \mathcal{H}$ is called bounded if there exists a $c \in \mathbb{R}$ such that $\|\hat{X}|\psi\rangle\| \leq c\|\psi\|$ for every $|\psi\rangle \in \text{dom } \hat{X}$. Otherwise $\hat{X}$ is said to be unbounded. The least of these upper bounds gives the norm of the operator. Using the properties of the norm in $\mathcal{H}$ one gets the following equivalent definitions of the norm of a linear operator:

$$\|\hat{X}\| := \sup_{\|\psi\| > 0} \frac{\|\hat{X}|\psi\rangle\|}{\|\psi\|} = \sup_{\|\psi\| = 1} \|\hat{X}|\psi\rangle\| = \sup_{\|\psi\| \leq 1} \|\hat{X}|\psi\rangle\| \quad \forall |\psi\rangle \in \text{dom } \hat{X}.$$
For a weak Cauchy sequence one gets a weak continuous operator. An operator \( \hat{X} \) on \( \text{dom} \hat{X} \) is simply called continuous if \( \hat{X} \) is continuous for every \( |\psi\rangle \in \text{dom} \hat{X} \). A linear operator \( \hat{X} \) on \( \text{dom} \hat{X} \) is either continuous everywhere on \( \text{dom} \hat{X} \) or nowhere! Another striking property of linear operators is the following: A linear operator is continuous if and only if it is bounded!

Though continuity allows to exchange an operator with a limes operation it is a severe restriction on an operator—most physical relevant operators aren’t continuous. Nevertheless, the physical interesting operators have the following property, which is a weakening of continuity: An operator \( \hat{X} \) on \( \text{dom} \hat{X} \subseteq \mathcal{H} \) is called closed \( \iff \) every sequence \( (|\psi_n\rangle)_n \subseteq \text{dom} \hat{X} \) with \( s\text{-lim}_{n \to \infty} |\psi_n\rangle = |\psi\rangle \) and \( s\text{-lim}_{n \to \infty} \hat{X}|\psi_n\rangle = |\psi'\rangle \) implies that \( |\psi\rangle \in \text{dom} \hat{X} \) and \( \hat{X}|\psi\rangle = |\psi'\rangle \).

A continuous linear operator \( \hat{X} \) on \( \text{dom} \hat{X} \) is closed if and only if \( \text{dom} \hat{X} \) is closed. A linear and bounded (hence continuous) operator can always be closed (Großmann [35]).

### 1.2.2 Classification of Linear Operators

Linear operators can be classified according to the scheme in figure 1.1. The physical most relevant operators are unitary and self-adjoint operators. Some properties of the various classes are given in the following sections.

![Figure 1.1: Schematic classification of linear operators on a Hilbert space \( \mathcal{H} \)](image)
Bounded Linear Operators

Every bounded operator \( \hat{X} \) with \( \operatorname{dom} \hat{X} \subset \mathcal{H} \) can be extended to a bounded operator \( \hat{Y} \) with \( \operatorname{dom} \hat{Y} = \mathcal{H} \), though this extension might not be unique. (For \( \operatorname{dom} \hat{X} \) dense in \( \mathcal{H} \) the extension is unique.) Hence for bounded operators \( \operatorname{dom} \hat{X} = \mathcal{H} \) can always be assumed.\(^8\)

Since a bounded operator \( \hat{X} \) is in particular continuous, operator and limes can be exchanged:

\[
\hat{X} \lim |\psi_n| = \lim \hat{X}|\psi_n| \quad \text{and} \quad \hat{X} \sum_{i=1}^{\infty} |\psi_i| = \sum_{i=1}^{\infty} \hat{X}|\psi_i|.
\]

The set of all bounded linear operators is also a Hilbert space. It is denoted by \( L(\mathcal{H}, \mathcal{H}) \) or shorter by \( L(\mathcal{H}) \).\(^9\)

A bounded operator \( \hat{X} : \operatorname{dom} \hat{X} = \mathcal{H} \rightarrow \mathcal{H} \) is called compact \( \iff \) every bounded set in \( \mathcal{H} \) is mapped into a compact set in \( \mathcal{H} \). A compact operator applied to a weak convergent sequence results in a strong convergent sequence.

Unbounded Operators

Almost the only general thing that can be said about unbounded linear operators is that they aren’t continuous anywhere on their domain. Though many physical relevant unbounded operators are closed.

Adjoint Operator

A very important concept for linear operators is the adjoint operator. For bounded operators its definition poses no problems. To generalize the definition to unbounded operators one has to assume that \( \operatorname{dom} \hat{X} \) is a dense linear subspace of \( \mathcal{H} \). Hence for unbounded operators a dense domain will always be assumed (see chapter 16 of Großmann [36]).\(^10\)

Every bounded operator \( \hat{X} \) on \( \mathcal{H} \) can be uniquely assigned the adjoint operator \( \hat{X}^\dagger \), such that \( \langle \phi | \hat{X} \psi \rangle = \langle \hat{X}^\dagger \phi | \psi \rangle \) for all \( |\phi\rangle, |\psi\rangle \in \mathcal{H} \). The adjoint operator is again a bounded linear operator and \( \|\hat{X}^\dagger\| = \|\hat{X}\| \). For \( \hat{X}, \hat{Y} \) bounded linear operators on \( \mathcal{H} \) also hold \( (\hat{X} \hat{Y})^\dagger = \hat{Y}^\dagger \hat{X}^\dagger \), \( \hat{X}^{\dagger\dagger} := (\hat{X}^\dagger)^\dagger = \hat{X} \) and \( (\alpha \hat{X} + \beta \hat{Y})^\dagger = \alpha^* \hat{X}^\dagger + \beta^* \hat{Y}^\dagger \) for all \( \alpha, \beta \in \mathbb{C} \).

If \( \hat{X} \) is an unbounded operator with a dense domain \( \operatorname{dom} \hat{X} \), one can assign to it uniquely\(^11\) the adjoint operator \( \hat{X}^\dagger \) on \( \operatorname{dom} \hat{X}^\dagger \), such that \( \langle \phi | \hat{X} \psi \rangle = \langle \hat{X}^\dagger \phi | \psi \rangle \) for all \( |\psi\rangle \in \operatorname{dom} \hat{X} \) and all \( |\phi\rangle \in \operatorname{dom} \hat{X}^\dagger \). \( \hat{X}^\dagger \) is again a linear operator, though \( \operatorname{dom} \hat{X}^\dagger \) need not be dense in \( \mathcal{H} \).

\( \hat{X}^\dagger \) is always a closed operator on \( \operatorname{dom} \hat{X}^\dagger \) (no matter whether \( \hat{X} \) is closed or not). If in addition \( \hat{X} \) is injective and \( \operatorname{ran} \hat{X} \) is a dense subspace of \( \mathcal{H} \) then \( (\hat{X}^{-1})^\dagger = (\hat{X}^\dagger)^{-1} \).

---

\(^8\) A bounded linear operator that is defined everywhere on \( \mathcal{H} \) is sometimes called linear-bounded.

\(^9\) The dual space \( \mathcal{H}' \) of \( \mathcal{H} \) is equal to \( L(\mathcal{H}, \mathbb{C}) \), the set of all bounded linear functionals from \( \mathcal{H} \) into the scalar field \( \mathbb{C} \).

\(^10\) The domain of bounded operators can always be assumed to be entire \( \mathcal{H} \). All the more it can be assumed to be dense in \( \mathcal{H} \). Hence in general theorems applying both to bounded and unbounded operators one always assumes a dense domain of an operator.

\(^11\) It is this uniqueness of the adjoint operator that enforces \( \operatorname{dom} \hat{X} \) to be a dense linear subspace of \( \mathcal{H} \) (see chapter 16 of Großmann [36]).
Isometric and Unitary Operators

A linear operator \( \hat{T} \colon \text{dom}\,\hat{T} = \mathcal{H}_1 \rightarrow \text{ran}\,\hat{T} = \mathcal{H}_2 \) between two Hilbert spaces\(^{12}\) is called isometric if \( \langle \hat{T}\phi|\hat{T}\psi \rangle = \langle \phi|\psi \rangle \) for all \( |\phi\rangle, |\psi\rangle \in \mathcal{H}_1 \) (i.e., \( \hat{T} \) preserves lengths and angles). From this follows that \( \|\hat{T}\| = 1 \), i.e., every isometric operator is bounded. According to the previous paragraph the adjoint operator \( \hat{T}^\dagger \) exists and is also bounded. In addition, every isometric operator has an inverse operator \( \hat{T}^{-1} \) that is also isometric (and hence bounded). On \( \text{ran}\,\hat{T} = \mathcal{H}_2 \) holds the relation \( \hat{T}^{-1} = \hat{T}^\dagger \) which is equivalent to \( \hat{T}^\dagger \hat{T} = \hat{T}^{-1} \hat{T} = I \) on \( \mathcal{H}_1 \). An isometric operator \( \hat{U} \) is called unitary if \( \text{dom}\,\hat{U} = \text{ran}\,\hat{U} = \mathcal{H} \). As for isometric operators \( \hat{U}^\dagger = \hat{U}^{-1} \) exist on \( \mathcal{H} \) and are unitary. Additionally, \( \hat{U}^\dagger \hat{U} = \hat{U} \hat{U}^\dagger = I \) on \( \mathcal{H} \). An important example of unitary operators are the time-evolution operators in quantum mechanics.

Symmetric Operators

A linear operator \( \hat{X} \) with a dense domain \( \text{dom}\,\hat{X} \subseteq \mathcal{H} \) is called symmetric if \( \langle \phi|\hat{X}\psi \rangle = \langle \hat{X}\phi|\psi \rangle \) for all \( |\phi\rangle, |\psi\rangle \in \text{dom}\,\hat{X} \). A symmetric operator has always an adjoint operator \( \hat{X}^\dagger \) and \( \hat{X} \subseteq \hat{X}^\dagger \). Moreover, \( \hat{X} \subseteq \hat{X}^{\dagger\dagger} \subseteq \hat{X}^\dagger \).

Self-Adjoint Operators

For bounded linear operators holds: A symmetric operator (sometimes also called hermitian) is always self-adjoint and vice versa. In general, a symmetric operator \( \hat{X} \) is called self-adjoint if \( \hat{X} = \hat{X}^\dagger \).

For a self-adjoint operator \( \hat{X} \), no matter if bounded or unbounded, always holds that \( \hat{X}^{\dagger\dagger} = \hat{X} \). Moreover, every self-adjoint operator is closed (since \( \hat{X}^\dagger \) is already closed). I.e., self-adjoint operators can always be exchanged with a limes operation. These two properties make self-adjoint operators so useful.\(^{13}\)

1.2.3 Some Remarks on Linear Operators

Since physicists insist on using their “bra-ket” notation even when no inner product is involved something has to be said about notation (see Cohen-Tannoudji et al. [20]). The bras and kets in this section shall always be elements of the appropriate domains.

As has already been said \( \langle \chi|\psi \rangle \) is another way of writing the inner product \( (|\chi\rangle, |\psi\rangle) \). Due to the properties of the inner product \( \langle \chi|\psi \rangle = \langle \psi|\chi \rangle^* \). The notation \( \hat{X}\psi \) shall mean applying the operator \( \hat{X} \) to the ket \( |\psi\rangle \): \( |\hat{X}\psi \rangle = \hat{X}|\psi\rangle \). Accordingly, \( |\lambda\psi\rangle \) is the product of the complex number \( \lambda \) with the ket \( |\psi\rangle \): \( |\lambda\psi\rangle = \lambda|\psi\rangle \). Since the bra \( \langle \phi | \) is the functional \( \langle |\phi \rangle, \cdot \rangle \) the symbol

\(^{12}\) Often, \( \mathcal{H}_2 \) is a subspace of \( \mathcal{H}_1 \).

\(^{13}\) The preface of Courant and Hilbert [22] gives an account of a meeting of the mathematician Friedrichs with Heisenberg; Friedrichs expressed the thankfulness of the mathematicians to one of the originators of quantum mechanics, a field that has brought force so much beautiful mathematics. Heisenberg agreed that this were true. Friedrichs answered that mathematics has paid some of this debt back to physics. Heisenberg stayed non-committal, so Friedrichs pointed out that it had been a mathematician, von Neumann, who had made clear the difference between a self-adjoint operator and one that is only symmetric. “What is the difference?” asked Heisenberg.
1.3 The Hilbert Spaces \( \ell^2 \) and \( L^2(\mathbb{R}^n) \)

The Hilbert spaces normally used in quantum mechanics are the space \( L^2(\mathbb{R}^n) \) and the norm of \( \psi \) is defined as the linear operator \( \langle \cdot | \cdot \rangle \) follows

\[
\langle \psi | \phi \rangle := \int_{\Omega} \phi^*(x) \psi(x) \, dx
\]

and the norm of \( |\phi\rangle \) by

\[
\| |\phi\rangle \| := \sqrt{\int_{\Omega} |\phi(x)|^2 \, dx}.
\]

Another often used space is the space \( \ell^2 \). The vectors of the Hilbert space \( \ell^2 \) are the infinite sequences \( |\cdot\rangle := \{a_i\}_{i=1}^\infty, a_i \in \mathbb{C} \), such that \( \sum_{i=1}^\infty |a_i|^2 < \infty \). The inner product is given by

\[
\langle \phi | \psi \rangle := \sum_{i=1}^\infty a_i^* b_i, \quad \text{where} \quad |\phi\rangle = \{a_i\}_{i=1}^\infty \text{ and } |\psi\rangle = \{b_i\}_{i=1}^\infty.
\]

The norm of \( |\phi\rangle \) is defined by

\[
\| |\phi\rangle \| := \sqrt{\sum_{i=1}^\infty |a_i|^2}.
\]

Both Hilbert spaces are isomorph \( \ell^2 \cong L^2 \). This is the mathematical reason that Heisenberg’s matrix algebra and Schrödinger’s wave functions give the same results.

1.4 The Postulates of Quantum Mechanics

At every time each quantum state is described completely by a ray in \( \mathcal{H} \). A ray is defined as the equivalence class of vectors \( ||\psi|| := \{c|\psi\rangle : |\psi\rangle \in \mathcal{H}, c \in \mathbb{C} \setminus \{0\} \} \) (see van Hees [89, 90]). To
If \( \hat{O} \) is a self-adjoint operator on \( \text{dom} \hat{O} \subseteq \mathcal{H} \). To every classical observable exits a quantum mechanical counterpart, while the reverse might not be the case (e.g., the spin of elementary particles or the parity). The only possible results of the measurement of an observable are elements of the spectrum of the assigned operator. The expectation value for the observable \( \hat{O} \) in the (normalized) state \( |\psi\rangle \) is given by (see section III.C.4 in [20])

\[
\langle O \rangle = \langle \psi | \hat{O} | \psi \rangle .
\] (1.4.1)

The dynamics of the quantum mechanical system is given by the Schrödinger equation

\[
i \hbar \frac{d}{dt} |\psi(t)\rangle = \hat{H} |\psi(t)\rangle .
\] (1.4.2)

### 1.5 Pure and Mixed States

A ket \( |\psi\rangle \) represents a pure state—or micro-state—of the system. It contains all the information that can be obtained from the system. Introducing the density operator \( \hat{\rho} = |\psi\rangle \langle \psi | \) the expectation value of the observable \( \hat{O} \) can be written as

\[
\langle O \rangle = \text{tr} \hat{\rho} \hat{O} ,
\] (1.5.1)

since \( |\psi\rangle \) and hence \( \hat{\rho} \) are normalized.

This concept of the density operator can be extended to systems where one only knows that the system has some probability \( p_k \) to be in a certain state \( \psi_k \). The various probabilities \( p_1, p_2, \ldots, p_k, \ldots \) are arbitrary on the condition that \( 0 \leq p_1, p_2, \ldots, p_k, \ldots \leq 1 \) and \( \sum_k p_k = 1 \). The density operator in this case is given by

\[
\hat{\rho} = \sum_k p_k |\psi_k\rangle \langle \psi_k | .
\] (1.5.2)

The expectation value of \( \hat{O} \) is again obtained via

\[
\langle O \rangle = \text{tr} \hat{\rho} \hat{O} .
\] (1.5.3)

If \( \hat{\rho} \) is unnormalized the formulas of the expectation values have to be divided by \( \text{tr} \hat{\rho} \).

---

14To be more precisely: the probability density amplitude.
15\( ab^* + a^* b = 2\Re(ab^*) \) and \( ab^* - a^* b = 2\Im(ab^*) \) for all \( a, b \in \mathbb{C} \).
2 Classical Systems

The last chapter showed how to describe statistical systems in quantum mechanics. Now the question arises, how the laws of classical statistical mechanics follow from the quantum mechanical description of large systems. [21, 24, 51, 64, 78, 84]

2.1 Phase Space and Micro-States

In classical mechanics, the state of a system of $N$ particles with $f$ degrees of freedom per particle is determined by specifying the $Nf$ (generalized) coordinates $q_k$ and the $Nf$ (generalized) momenta $p_k$ ($k = 1, \ldots, Nf$). Using these $q_k$ and $p_k$ as Cartesian coordinates, the state of the system can be described as a point in the $2Nf$-dimensional phase space $\Gamma$. As long as the internal degrees of freedom of the particles can be neglected, one has $f = 3$ degrees of freedom per particle. In this case, the phase space becomes $6N$-dimensional.

For such a classical system with $N$ particles, with three degrees of freedom per particle, the obvious observables are the positions $r_i$ ($i = 1, \ldots, N$) of the centers of mass, the momenta $p_i$ and, more generally, functions $f(r^N, p^N) = f(r_1, p_1, \ldots, r_N, p_N)$ of these $6N$ variables. A micro-state, i.e., a state in which “everything” about the system is known, is defined by specifying the values of all these $6N$ observables.

A classical macro-state, i.e., a state of the system in which the dynamical variables of the system are not completely known or prepared, is characterized by giving a probability law for the possible micro-states, that is, for the set of possible points in phase space. A classical macro-state is thus represented by a probability measure in phase space:

$$w(r^N, p^N) \, d\Gamma,$$  \hspace{1cm} (2.1.1)

where $w(r^N, p^N)$ is called density in phase space or phase density. The quantity in (2.1.1) is the probability that the representative point of the $N$ particles of the system is within the volume element $d\Gamma_N = dr^N dp^N = dr_1 dp_1 \cdots dr_N dp_N$ of phase space.

With this probability measure the expectation value for the classical observable $O(r^N, p^N)$ is given by

$$\langle O \rangle = \int_{\Gamma} O(r^N, p^N) w(r^N, p^N) \, d\Gamma.$$  \hspace{1cm} (2.1.2)

---

1 The $p_k$ are also called the conjugated momenta to the generalized coordinates $q_k$.

2 $\Gamma$ stands for “gas” [24, 84].
2.2 Classical Statistical Mechanics

In the following a Hamiltonian of the special form $H(r^N, p^N) = K(p^N) + U(r^N)$ will be used. $K(p^N)$ denotes the total kinetic and $U(r^N)$ the total potential energy of the $N$ particles. It will further be assumed that the potential energy is a pairwise additive function of the relative distances between the particles only. With this assumptions the Hamiltonian becomes

$$H(r^N, p^N) = \sum_{i=1}^{N} \frac{p_i^2}{2m} + \sum_{i=1}^{N} \sum_{j=i+1}^{N} U_{ij}(|r_i - r_j|),$$  \hspace{1cm} (2.2.1)

where $U_{ij}(|r_i - r_j|)$ gives the potential energy of the interaction between the particles $i$ and $j$. For a binary mixture of $N_1$ particles of substance 1 and $N_2$ particles of substance 2 ($N = N_1 + N_2$) the coordinates and the momenta of the particles will be labeled $r^{N_1} = r_1, \ldots, r_{N_1}$, $r^{N_2} = r_{N_1+1}, \ldots, r_{N_1+N_2}$, $p^{N_1} = p_1, \ldots, p_{N_1}$ and $p^{N_2} = p_{N_1+1}, \ldots, p_{N_1+N_2}$, respectively. The Hamiltonian (2.2.1) thus becomes

$$H(r^{N_1}, r^{N_2}, p^{N_1}, p^{N_2}) = \sum_{i=1}^{N_1} \frac{p_i^2}{2m_i} + \sum_{i=N_1+1}^{N_1+N_2} \frac{p_i^2}{2m_2} + U(r^{N_1}, r^{N_2}).$$  \hspace{1cm} (2.2.2)

The potential energy $U(r^{N_1}, r^{N_2}) = U^{1\rightarrow1}(r^{N_1}) + U^{1\rightarrow2}(r^{N_1}, r^{N_2}) + U^{2\rightarrow2}(r^{N_2})$ is the sum of interactions between particles of substance 1

$$U^{1\rightarrow1}(r^{N_1}) = \sum_{i=1}^{N_1-1} \sum_{j=i+1}^{N_1} U^{1\rightarrow1}_{ij}(|r_i - r_j|),$$

interactions between particles of substance 1 and substance 2

$$U^{1\rightarrow2}(r^{N_1}, r^{N_2}) = \sum_{i=1}^{N_1} \sum_{j=N_1+1}^{N_1+N_2} U^{1\rightarrow2}_{ij}(|r_i - r_j|),$$

and interactions between particles of substance 2

$$U^{2\rightarrow2}(r^{N_2}) = \sum_{i=N_1+1}^{N_1+N_2-1} \sum_{j=i+1}^{N_1+N_2} U^{2\rightarrow2}_{ij}(|r_i - r_j|).$$

The two-particle interaction potentials $U_{ij}$ will in general be different in the three sums above.

If the interaction between substance (subsystem) 1 and substance (subsystem) 2 could be neglected (e.g., for ideal gas mixtures), the partition functions of the total system would be the product of the partition functions of the subsystems. Since in general $U^{1\rightarrow2}(r^{N_1}, r^{N_2})$ will not vanish this factorization cannot be performed (see also the quotation at the beginning of chapter 4).

2.2.1 Canonical Ensemble

In the canonical ensemble the particle number $N$, the volume $V$ and the temperature $T$ of a system are fixed. Hence the canonical ensemble is also called $NVT$ ensemble.
2.2 Classical Statistical Mechanics

**Pure Systems**

The partition function in the canonical ensemble is given by

\[ Q(N, V, T) = \frac{1}{(2\pi \hbar)^{3N}} \frac{1}{N!} \int_{\Gamma} e^{-\beta H(r^N, p^N)} \, dr^N \, dp^N. \]  

(2.2.3)

Classically, the particles of the system are distinguishable. Hence exchanging two particles gives a different micro-state of the system, though these two micro-states can’t be distinguished by macroscopical measurements. For \( N \) particles there are \( N! \) possibilities for such exchanges. To take these exchanges into account the partition function is multiplied by \( 1/N! \).

With the Hamiltonian from (2.2.1) one obtains

\[ Q(N, V, T) = \frac{1}{N! A^{3N}} \int_V e^{-\beta U(r^N)} \, dr^N. \]  

(2.2.5)

For an ideal gas \( U(r^N) \equiv 0 \) and hence the integral becomes \( V^N \), resulting in

\[ Q_{id}(N, V, T) = \frac{V^N}{N! A^{3N}} = \frac{V^N}{N!} \left( \frac{mk_B T}{2\pi \hbar^2} \right)^{3N/2}. \]  

(2.2.6)

The partition function can be split into a product \( Q(N, V, T) = Q_{id}(N, V, T)Q_{ex}(N, V, T) \), where

\[ Q_{ex}(N, V, T) = \frac{1}{V^N} \int_V e^{-\beta U(r^N)} \, dr^N \]

gives the deviation from an ideal gas due to the intermolecular interactions. With the definition of the configurational part of the partition function \( Q(N, V, T) \)

\[ Z(N, V, T) = \int_V e^{-\beta U(r^N)} \, dr^N \]  

(2.2.7)

it follows that \( Q_{ex} = Z/V^N \).
Mixtures

For binary mixtures the partition function becomes

\[
Q(N_1, N_2, V, T) = \frac{1}{(2\pi \hbar)^3 N_1! N_2!} \int e^{-\beta H(r^{N_1}, r^{N_2}, p^{N_1}, p^{N_2})} \, dr^{N_1} dr^{N_2} dp^{N_1} dp^{N_2}.
\]

Here the partition function is multiplied by \(1/(N_1! N_2!)\). As explained before exchanging two particles gives different micro-states that can’t be distinguished by macroscopical measurements. But this is only true for particle exchanges of the same species: If one particle of substance 1 is exchanged with one particle of substance 2 it is in principle possible to measure macroscopically differences. Therefore the correction factor can’t be \(1/(N_1 + N_2)!\).

With a Hamiltonian of the form (2.2.2) one obtains, in complete analogy with the case of the pure system,

\[
Q(N_1, N_2, V, T) = \frac{1}{N_1!} \Lambda^{3 N_1} \frac{1}{N_2!} \Lambda^{3 N_2} \int_V e^{-\beta U(r^{N_1}, r^{N_2})} \, dr^{N_1} dr^{N_2},
\]

(2.2.8)

where \(\Lambda_1\) and \(\Lambda_2\) are the thermal de Broglie wavelengths for substance 1 and substance 2, respectively. If the interaction \(U^{1 \leftrightarrow 2}(r^{N_1}, r^{N_2})\) between the subsystems can be neglected, the partition function becomes a product of the partition functions of the subsystems (as for ideal gases).

Expectation Values

Let \(O(r^N, p^N)\) be an observable of the system. Then the expectation value of \(O\) is given by

\[
\langle O \rangle = \frac{1}{Z} \frac{1}{(2\pi \hbar)^3 N!} \int e^{-\beta H(r^N, p^N)} O(r^N, p^N) \, dr^N dp^N
\]

(2.2.9)

(the formula for mixtures is analogous). The observable \(O(r^N, p^N)\) is therefore averaged over the probability density \(w(r^N, p^N) \propto e^{-\beta H(r^N, p^N)}\). In this general case no simplifications can be performed. In the following two special cases are studied where a considerable simplification can be achieved.

Let \(O = O(r^N)\) depend only on the positions of the \(N\) particles and suppose that the Hamiltonian is of the form (2.2.1). In this case the integration over phase space can be factored into an integration over the momenta and an integration over the coordinates. Hence (2.2.9) becomes

\[
\langle O \rangle = \frac{V^N}{Q_{id} Z} \frac{1}{(2\pi \hbar)^3 N!} \int_{-\infty}^{\infty} e^{-\beta \sum p_i^2/2m} \, dp^N \int_V e^{-\beta U(r^N)} O(r^N) \, dr^N.
\]

Thus the expectation value of \(O\) is given by an integration over the configurational space alone

\[
\langle O \rangle = \frac{1}{Z} \int_V e^{-\beta U(r^N)} O(r^N) \, dr^N.
\]

(2.2.10)
The probability density here is \( w(r^N) \propto e^{-\beta U(r^N)} \).

The other special case is \( O = O(p^N) \). Splitting the integration over the phase space and reducing as before yields

\[
\langle O \rangle = \left( \frac{\beta}{2\pi m} \right)^{3N/2} \int_{-\infty}^{\infty} e^{-\beta \sum_i p_i^2 / 2m} O(p^N) \, dp^N
\]

and the probability density is given by \( w \propto e^{-\beta \sum_i p_i^2 / 2m} \). Normally these integrals can be solved analytically. An important example is \( O = p_i^2 \) \((j = 1, \ldots, 3N)\). Using the properties of the exponential function one gets

\[
\langle p_i^2 \rangle = \left( \frac{\beta}{2\pi m} \right)^{3N/2} \int_{-\infty}^{\infty} e^{-\beta \sum_{i\neq j} p_i^2 / 2m} \, dp_3^{3N-1} \int_{-\infty}^{\infty} e^{-\beta p_j^2 / 2m} p_j^2 \, dp_j.
\]

The first integral gives \( \left( \int_{-\infty}^{\infty} e^{-\beta p^2 / 2m} \, dp \right)^{3N-1} = (2\pi m / \beta)^{(3N-1)/2} \) and the second integral yields \( \sqrt{\pi / 2(2m / \beta)^{3/2}} \). Collecting these three factors the expectation value becomes \( \langle p_j^2 \rangle = \langle p_i^2 / 2m \rangle = 3/2k_B T \) \((i = 1, \ldots, N)\) which is the well-known equipartition theorem.

One important observable is the pressure \( p \) of the system. It is calculated using the virial of the system, as explained by Frenkel and Smit [29].

### 2.2.2 Isothermal-Isobaric Ensemble

#### Pure Systems

The transition from the canonical to the isothermal-isobaric ensemble is given by the proper Laplace transformation (see section A.3) [29, 76]

\[
Q(N, p, T) = \beta p \int_0^\infty e^{-\beta pV} Q(N, V, T) \, dV.
\]

The dimension of the phase space has therefore been increased to \( 6N + 1 \). Inserting the canonical partition function from (2.2.3) yields the isothermal-isobaric partition function

\[
Q(N, p, T) = \frac{1}{(2\pi \hbar)^{3N}} \frac{\beta p}{N!} \int_0^\infty \int_{F(V)} e^{-\beta [H(r^N, p^N) + pV]} \, dr^N dp^N \, dV.
\]

In the two preceding formulas the shape of the volume is assumed to be constant. In the derivation of (2.2.12) by Frenkel and Smit [29] the shape of the volume is always a cubical box. In the integration over the volume this cubical box is then inflated without changing its shape. To avoid difficulties with changes of the shape of the volume in the following only cubical bounding boxes will be assumed. Furthermore, to simplify the following formulas one introduces scaled coordinates \( s^N \) by

\[
r_i = V^{1/3} s_i \quad (i = 1, \ldots, N).
\]
The integrations $\int V \ldots dr^N$ are thereby transformed into integrations $V^N \int \Box_1 \ldots ds^N$ over cubic boxes $\Box_1$ of unit volume.

With the Hamiltonian from (2.2.1) the integration over the momenta can again be factored out and the partition function becomes

$$Q(N, p, T) = \frac{\beta p}{N! A^{3N}} \int_0^\infty V^N e^{-\beta pV} \int \Box_1 e^{-\beta U(V^{1/3} s^N)} ds^N dV.$$  \hspace{1cm} (2.2.14)

An illustrative derivation of this equation based on simple statistical mechanics considerations can be found in section 5.4.1 of [29]. Since in general the integrations over the $r^N = V^{1/3} s^N$ depend on the volume the partition function cannot be further simplified. In the special case of an ideal gas ($U(r^N) \equiv 0$) one obtains

$$Q_{id}(N, p, T) = \frac{\beta p}{N! A^{3N}} \int_0^\infty V^N e^{-\beta pV} dV.$$  \hspace{1cm} (2.2.15)

The integral over the volume is a standard integral and has the value $N! / (\beta p)^{N+1}$. Hence the partition function for an ideal gas in the isothermal-isobaric ensemble becomes

$$Q_{id}(N, p, T) = \left( \frac{k_B T}{\beta p} \right)^N \left( \frac{mk_B T}{2\pi \hbar^2} \right)^{3N/2}.$$  \hspace{1cm} (2.2.16)

The partition function can again be split into a product $Q(N, p, T) = Q_{id}(N, p, T)Q_{ex}(N, p, T)$ with the excess part given by

$$Q_{ex}(N, p, T) = (\beta p)^{N+1} \frac{1}{N!} \int_0^\infty V^N e^{-\beta pV} \int \Box_1 e^{-\beta U(V^{1/3} s^N)} ds^N dV.$$  \hspace{1cm} (2.2.17)

Mixtures

With the Hamiltonian from (2.2.2) one obtains as for the canonical ensemble

$$Q(N_1, N_2, p, T) = \frac{\beta p}{N_1! A_1^{3N_1} N_2! A_2^{3N_2}} \int_0^\infty V^{N_1+N_2} e^{-\beta pV \times}$$

$$\times \int \Box_1 e^{-\beta U(V^{1/3} s^N_1, V^{1/3} s^N_2)} ds^N_1 ds^N_2 dV.$$  \hspace{1cm} (2.2.18)

Again for vanishing $U^{1-s_2}(r^{N_1}, r^{N_2})$ the partition function can be split into a product of the partition functions for the pure systems.
2.2 Classical Statistical Mechanics

Expectation Values

As for the canonical ensemble the expectation value of an observable $O(r^N, p^N)$ is given by

$$\langle O \rangle = \frac{1}{Q} \frac{1}{(2\pi \hbar)^{3N}} \frac{\beta p}{N!} \int_0^\infty \int_{\Gamma(V)} e^{-\beta[H(r^N, p^N) + pV]} O(r^N, p^N) \, dr^N \, dp^N \, dV. \quad (2.2.18)$$

The formula for mixtures is analogous. Since the actual form of $O(r^N, p^N)$ is in general not known these integrals cannot be further simplified. Nevertheless, one can again study two important cases.

Let $O = O(r^N)$ depend only on the coordinates of the $N$ particles and suppose that the Hamiltonian of the system is of the form (2.2.1). Then the integrals over the momenta and the multiplicative factors cancel each other as in the canonical ensemble and the expectation value in (2.2.18) becomes

$$\langle O \rangle = \frac{1}{Z} \int_0^\infty V^N e^{-\beta p V} \int_{\Sigma_{1}} e^{-\beta U(V^{1/3} s^N)} O(V^{1/3} s^N) \, ds^N \, dV. \quad (2.2.19)$$

The probability density in this case is given by $w(V, s^N) \propto V^N e^{-\beta[U(V^{1/3} s^N) + pV]}$.

If the observable depends only on the $N$ momenta of the particles the configurational parts of the integrals cancel and the expectation value is given by the same formula as in the canonical ensemble

$$\langle O \rangle = \left( \frac{\beta}{2\pi m} \right)^{3N/2} \int_{-\infty}^\infty e^{-\beta \sum p_i^2 / 2m} O(p^N) \, dp^N. \quad (2.2.20)$$

From this it follows that the equipartition theorem for the kinetic energy is also valid in the isothermal-isobaric ensemble as was to be expected.
3 Path-Integral Formalism

3.1 Mathematical Preliminaries

Ignoring possible difficulties with the domains of the involved operators and the convergence of
the Taylor series, one has the following results: Using the Taylor expansion of the exponential
function, for two operators \( \hat{X} \) and \( \hat{Y} \) one has on the one hand
\[
e^{b\hat{X} + b\hat{Y}} = \hat{I} + (\hat{X} + \hat{Y}) + \frac{1}{2}(\hat{X} + \hat{Y})^2 + \frac{1}{6}(\hat{X} + \hat{Y})^3 + \cdots
\]
\[
e^{b\hat{X}} e^{b\hat{Y}} = (\hat{I} + \frac{1}{2}\hat{X}^2 + \frac{1}{6}\hat{X}^3 + \cdots)(\hat{I} + \frac{1}{2}\hat{Y}^2 + \frac{1}{6}\hat{Y}^3 + \cdots)
\]
\[
e^{b\hat{X} + b\hat{Y}} = \hat{I} + (\hat{X} + \hat{Y}) + \frac{1}{2}(\hat{X}^2 + \hat{X}\hat{Y} + \hat{Y}\hat{X} + \hat{Y}^2) + \frac{1}{6}(\hat{X}^3 + \hat{X}^2\hat{Y} + \hat{X}\hat{Y}^2 + \hat{Y}\hat{X}^2 + \hat{Y}\hat{X}\hat{Y} + \hat{Y}^2\hat{X} + \hat{Y}^3) + \cdots
\] (3.1.1)

and on the other hand
\[
e^{b\hat{X}} e^{b\hat{Y}} = (\hat{I} + \frac{1}{2}\hat{X}^2 + \frac{1}{6}\hat{X}^3 + \cdots)(\hat{I} + \frac{1}{2}\hat{Y}^2 + \frac{1}{6}\hat{Y}^3 + \cdots)
\]
\[
e^{b\hat{X} + b\hat{Y}} = \hat{I} + (\hat{X} + \hat{Y}) + \frac{1}{2}(\hat{X}^2 + 2\hat{X}\hat{Y} + \hat{Y}^2) + \frac{1}{6}(\hat{X}^3 + 3\hat{X}^2\hat{Y} + 3\hat{X}\hat{Y}^2 + \hat{Y}^3) + \cdots
\] (3.1.2)

The difference is given by
\[
e^{b\hat{X} + b\hat{Y}} = e^{b\hat{X}} e^{b\hat{Y}} - \frac{1}{2}(\hat{X}\hat{Y} - \hat{Y}\hat{X})
\]
\[
e^{b\hat{X}} e^{b\hat{Y}} = (\hat{I} + \frac{1}{2}\hat{X}^2 + \frac{1}{6}\hat{X}^3 + \cdots)(\hat{I} + \frac{1}{2}\hat{Y}^2 + \frac{1}{6}\hat{Y}^3 + \cdots)
\]
\[
e^{b\hat{X}} e^{b\hat{Y}} = \hat{I} - \frac{1}{2}(2\hat{X}\hat{Y} + 2\hat{X}^2 - \hat{X}\hat{Y}\hat{X} - \hat{Y}\hat{X}^2 - \hat{Y}^2\hat{X} - \hat{X}\hat{Y}) - \cdots
\] (3.1.3)

So if \([\hat{X}, \hat{Y}] = 0\), i.e. \( \hat{X} \) and \( \hat{Y} \) commute, one obtains
\[
e^{b\hat{X} + b\hat{Y}} = e^{b\hat{X}} e^{b\hat{Y}}
\]

Since multiplying a self-adjoint operator by a real number gives again a self-adjoint operator, for non-commuting
operators and for large \( P \) equation (3.1.3) implies that
\[
e^{(b\hat{X} + b\hat{Y})/P} = e^{\hat{X}/P} e^{\hat{Y}/P} - \frac{1}{2P^2} [\hat{X}, \hat{Y}] + O(P^{-3})
\]

Trotter showed in [88] that under certain conditions (\( \hat{X} \) and \( \hat{Y} \) self-adjoint and a little bit
more) holds
\[
e^{\hat{X} + \hat{Y}} = \lim_{P \to \infty} \left( e^{\hat{X}/P} e^{\hat{Y}/P} \right)^P
\] (3.1.4)

so that for large enough Trotter numbers \( P \)
\[
e^{(\hat{X} + \hat{Y})/P} \approx e^{\hat{X}/P} e^{\hat{Y}/P}
\] (3.1.5)
3.2 Canonical Ensemble

If the density matrix of a system were known, one could calculate the expectation values of all observables of the system. Unfortunately, it is impossible to calculate the density matrix analytically for more than the simplest systems.

In the canonical ensemble the normalized density operator is given by

$$\hat{\rho} = \frac{e^{-\beta \hat{H}}}{Q(N,V,T)}, \quad (3.2.1)$$

where the partition function $Q(N,V,T)$ is the trace

$$Q(N,V,T) = \text{tr} e^{-\beta \hat{H}}. \quad (3.2.2)$$

3.2.1 The Density Operator for a Free Particle

One of the simplest systems is a free particle in one dimension. The Hamilton operator for a free particle of mass $m$ in one dimension is given by

$$\hat{H}_0 = \frac{\hat{p}^2}{2m}. \quad (3.2.3)$$

As is shown, e.g. in Feynman [26], the unnormalized density matrix in the coordinate representation is given by

$$\langle x'|e^{-\beta \hat{H}_0}|x \rangle = \rho_0(x', x; \beta) = \sqrt{\frac{m}{2\pi \hbar^2 \beta}} \exp \left[ -\frac{m(x' - x)^2}{2\hbar^2 \beta} \right]. \quad (3.2.4)$$

For a free particle in three dimensions one has analogously

$$\langle r'|\rho_0|r \rangle = \rho_0(r', r; \beta) = \left( \frac{m}{2\pi \hbar^2 \beta} \right)^{3/2} \exp \left[ -\frac{m(r' - r)^2}{2\hbar^2 \beta} \right].$$

3.2.2 One Particle in One Dimension

In the following only one particle in one dimension will be studied. The extension to systems with more particles and dimensions will be treated in the next section. The Hamiltonian $\hat{H}$ shall be of the form $\hat{H} = \hat{H}_0 + U(x)$.

Factorization of the Density Operator

Since $\hat{H}$ commutes with itself it follows for the density matrix

$$\rho(x'', x; \beta) = \sum_n \phi_n^*(x'') e^{-\beta \hat{H}} \phi_n(x)$$

$$= \sum_n \phi_n^*(x'') e^{-\frac{\beta \hat{H}}{2}} e^{-\frac{\beta \hat{H}}{2}} \phi_n(x),$$
3.2 Canonical Ensemble

where \( \{ |\phi_n\rangle \} \) is some Hilbert basis. In this basis one can write

\[
e^{-\frac{\beta}{2} \hat{H}} \phi_n(x) = \sum_m c_m \phi_m(x)
\]

with

\[
c_m = \int \phi_m^*(x') e^{-\frac{\beta}{2} \hat{H}} \phi_n(x') \, dx'.
\]

Inserting this expansion into the formula for \( \rho(x'', x; \beta) \) yields

\[
\rho(x'', x; \beta) = \sum_n \phi_n^*(x'') \sum_m \left[ e^{-\frac{\beta}{2} \hat{H}} \phi_m(x) \right] \left[ e^{-\frac{\beta}{2} \hat{H}} \phi_n(x') \right] \, dx' \\
= \left\{ \sum_n \phi_n^*(x'') \left[ e^{-\frac{\beta}{2} \hat{H}} \phi_n(x') \right] \sum_m \phi_m^*(x') \left[ e^{-\frac{\beta}{2} \hat{H}} \phi_m(x) \right] \, dx' \right\} \rho(x'', x; \frac{\beta}{2}) + \rho(x', x; \frac{\beta}{2})
\]

and hence

\[
\rho(x'', x; \beta) = \int \rho(x'', x'; \frac{\beta}{2}) \rho(x', x; \frac{\beta}{2}) \, dx'.
\] (3.2.5)

Now, the density matrix at temperature \( \beta \) has been split into two density matrices at \( \beta/2 \), i.e., twice the temperature. Applying the same procedure again one gets four density matrices at \( \beta/4 \) and so on. Writing \( x^{(0)}, x^{(1)}, x^{(2)}, \ldots \) instead of \( x, x', x'', \ldots \) one obtains

\[
\rho(x^{(P)}, x^{(0)}; \beta) = \int \rho(x^{(P)}, x^{(P-1)}; \frac{\beta}{P}) \cdots \rho(x^{(2)}, x^{(1)}; \frac{\beta}{P}) \rho(x^{(1)}, x^{(0)}; \frac{\beta}{P}) \, dx^{(1)} \cdots dx^{(P-1)},
\] (3.2.6)

where \( P \) is again the Trotter number. According to Feynman [26] the interpretation of the above expression is as follows: The particle travels in imaginary time \( \beta \hbar \) from \( x^{(0)} \) to \( x^{(P)} \) through a series of intermediate steps, \( x^{(1)}, x^{(2)}, \ldots, x^{(P-1)} \), which define a “path” (see figure 3.1). The total amplitude \( \rho(x^{(P)}, x^{(0)}; \beta) \) for the particle to begin at \( x^{(0)} \) and end up at \( x^{(P)} \) is given by a sum over all possible paths, that is, for all possible values of the intermediate positions \( x^{(k)} \). Hence (3.2.6) is called a path-integral.

The density matrices at \( \beta/P \) are given by

\[
\rho\left( x^{(k+1)}, x^{(k)}; \frac{\beta}{P} \right) = \sum_n \phi_n^*(x^{(k+1)}) e^{-\frac{\beta}{P} \hat{H}} \phi_n(x^{(k)}) \\
= \sum_n \phi_n^*(x^{(k+1)}) e^{-\frac{\beta}{P} [\hat{H}_0 + U(x^{(k)})]} \phi_n(x^{(k)}).
\]

Sometimes one uses \( (U(x^{(k)}) + U(x^{(k+1)}))/2 \) instead of \( U(x^{(k)}) \). For large enough Trotter numbers \( P \) follows from section 3.1 that

\[
e^{-\frac{\beta}{P} [\hat{H}_0 + U(x^{(k)})]} \approx e^{-\frac{\beta}{P} \hat{H}_0} e^{-\frac{\beta}{P} U(x^{(k)})} \approx e^{-\frac{\beta}{P} U(x^{(k)})} e^{-\frac{\beta}{P} \hat{H}_0}.
\]
Using this approximation the individual density matrices in (3.2.6) may be written as
\[
\rho(x^{(k+1)}, x^{(k)}; \beta) \approx \sum_n \phi_n^*(x^{(k+1)}) e^{-\frac{\beta}{P} U(x^{(k)})} e^{-\frac{\beta}{P} \tilde{H}_0 \phi_n(x^{(k)})} = \exp\left[ -\frac{\beta}{P} U(x^{(k)}) \right] \rho_0(x^{(k+1)}, x^{(k)}; \beta). 
\]

 Inserting the density matrix for a free particle from (3.2.4) yields
\[
\rho(x^{(k+1)}, x^{(k)}; \beta) \approx \sqrt{\frac{mP}{2\pi \hbar^2 \beta}} \int \exp\left[ -\frac{\beta}{P} \sum_{k=0}^{P-1} \frac{mP^2}{2\hbar^2 \beta^2} (x^{(k+1)} - x^{(k)})^2 \right] \rho_0(x^{(k+1)}, x^{(k)}; \beta). 
\] (3.2.7)

and together with (3.2.6) the complete density matrix at \( \beta \) becomes
\[
\rho(x^{(P)}, x^{(0)}; \beta) \approx \left( \frac{mP}{2\pi \hbar^2 \beta} \right)^{P/2} \int \exp\left[ -\frac{\beta}{P} \sum_{k=0}^{P-1} \frac{mP^2}{2\hbar^2 \beta^2} (x^{(k+1)} - x^{(k)})^2 \right] \times \exp\left[ -\frac{\beta}{P} \sum_{k=0}^{P-1} U(x^{(k)}) \right] dx^{(1)} \cdots dx^{(P-1)}. 
\] (3.2.8)

For \( P \to \infty \) this approximation becomes exact.

For the partition function \( Q(\beta) = \text{tr} \hat{\rho} = \int \rho(x, x; \beta) dx \), i.e., integration over \( x^{(P)} \) with the constraint \( x^{(k+P)} = x^{(k)} \), one obtains
\[
Q(\beta) \approx \left( \frac{mP}{2\pi \hbar^2 \beta} \right)^{P/2} \int \exp\left\{ -\frac{\beta}{P} \sum_{k=1}^{P} \frac{mP^2}{2\hbar^2 \beta^2} (x^{(k+1)} - x^{(k)})^2 + \sum_{k=1}^{P} U(x^{(k)}) \right\} dx^{(1)} \cdots dx^{(P-1)} dx^{(P)}. 
\] (3.2.9)
3.2 Canonical Ensemble

3.2.3 Classical Isomorphism

Equation (3.2.9) is similar to the partition function of $P$ classical particles in the effective potential

$$U_{\text{eff}}(x^{(1)}, \ldots, x^{(P)}) = \frac{mP}{2\hbar^2\beta^2} \sum_{k=1}^{P} (x^{(k+1)} - x^{(k)})^2 + \frac{1}{P} \sum_{k=1}^{P} U(x^{(k)}).$$

The effective potential may be visualized as that describing a “necklace” or ring-polymer of $P$ beads, each of which interacts with its neighbors through harmonic forces and with $(1/P)\text{th}$ of the external field [55]. This classical isomorphism is shown in figure 3.2. The “spring constant” between two consecutive beads of the ring-polymer is $mP/(\hbar^2\beta^2)$. The higher the quantum particle classical equivalent Trotter number the more stiff the “springs” become, while the effect of the external potential decreases with $1/P$.

According to Li and Broughton [55] the partition function in (3.2.9) is accurate to order $(\beta/P)^2$. Approximations of higher order can be found in [23, 31, 55].

3.2.4 $N$ particles in Three Dimensions

Analogous to one particle in one dimension one obtains for the partition function in Boltzmann statistics for $N$ particles in three dimensions

$$Q(N, V, T) \approx \frac{1}{N!} \left( \frac{mP}{2\pi \hbar^2 \beta} \right)^{3NP/2} \int \exp \left\{-\frac{\beta}{P} \sum_{i=1}^{N} \sum_{k=1}^{P} \frac{mP^2}{2\hbar^2\beta^2} (r^{(k+1)}_i - r^{(k)}_i)^2 + \sum_{k=1}^{P} U(r^{(k)}_1, \ldots, r^{(k)}_N) \right\} dr^{(1)}_1 \cdots dr^{(P)}_1 \cdots dr^{(1)}_N \cdots dr^{(P)}_N. \quad (3.2.10)$$

If the external potential is pairwise additive as in (2.2.1) only beads with the same index $(k)$ interact with each other through the external potential $U(r^{(k)}_1, \ldots, r^{(k)}_N)$. In this case the calculation effort is only $P$-times that for a classical system.
3 Path-Integral Formalism

Expectation Values

If the observable $O$ depends only on the coordinates of the particles its expectation value is given by

$$\langle O \rangle \approx \frac{1}{Q N!} \left( \frac{m P}{2 \pi \hbar^2 \beta} \right)^{3NP/2} \int \exp \left\{ -\frac{\beta}{P} \left[ \sum_{i=1}^N \sum_{k=1}^P \frac{m P^2}{2 \hbar^2 \beta^2} (r_{i}^{(k+1)} - r_{i}^{(k)})^2 + \sum_{k=1}^P U(r_1^{(k)}, \ldots, r_N^{(k)}) \right] \right\} \prod_{k=1}^P \frac{1}{P} O(r_1^{(k)}, \ldots, r_N^{(k)}) \, dr_1^{(1)} \cdots dr_N^{(1)} \cdots dr_1^{(P)} \cdots dr_N^{(P)}. \quad (3.2.11)$$

The expectation value of the kinetic energy is obtained by

$$\langle K \rangle \approx \frac{3NP}{2\beta} - \frac{1}{P} \left( \sum_{i=1}^N \sum_{k=1}^P \frac{m P^2}{2 \hbar^2 \beta^2} (r_{i}^{(k+1)} - r_{i}^{(k)})^2 \right). \quad (3.2.12)$$

Hence the kinetic energy is the classical result $3NP/(2\beta)$ as obtained from the equipartition theorem minus the energy of the “springs”. Since in this formulation the kinetic energy is the difference of about equal terms it is called “crude energy estimator”.

3.3 Isothermal-Isobaric Ensemble

Due to the classical isomorphism the transition from the canonical to the isothermal-isobaric ensemble is as for the classical case. But now, switching to scaled coordinates gives a factor of $V^{NP}$ instead of $V^N$ in the configurational integral.
4 Thermodynamics

If a system is very weakly coupled to a heat bath at a given “temperature”, if the coupling is indefinite or not known precisely, if the coupling has been on for a long time, and if all the “fast” things have happened and all the “slow” things not, the system is said to be in thermal equilibrium.

For instance, an enclosed gas placed in a heat bath will eventually erode its enclosure; but this erosion is a comparatively slow process, and sometime before the enclosure is appreciably eroded, the gas will be in thermal equilibrium.

– R. P. Feynman

4.1 Partition Functions and Thermodynamic Potentials

The previous chapters showed that for every system exists a partition function that contains all the information that can be retrieved from the system. This chapter shows how these partition functions are connected to the more axiomatic thermodynamical potentials.

4.1.1 Entropy and Inner Energy

For a system in the microcanonical ensemble the particle number $N$, the volume $V$ and the inner energy $E$ are fixed (such a system is called “closed” or “insulated”). The thermodynamical potential for such a system is the entropy $S$. It is connected to the microcanonical partition function $Q(N,V,E)$ via

$$ S(N,V,E) = k_B \ln Q(N,V,E). \quad (4.1.1) $$

For a closed system (a system under adiabatic-isochoric conditions) in thermal equilibrium the entropy becomes a maximum. Via the laws of thermodynamics only the total differential of the entropy can be defined by

$$ dS = \frac{1}{T} dE + \frac{p}{T} dV - \mu \frac{1}{T} dN - \cdots, \quad (4.1.2) $$

where $\mu$ is the chemical potential of the system. From the above equation one obtains $1/T = \partial S/\partial E$, $p/T = \partial S/\partial V$ and $-\mu/T = \partial S/\partial N$. It is often useful to use the inner energy $E = E(N,V,S)$ instead of the entropy. In this case holds

$$ dE = TdS - pdV + \mu dN + \cdots $$

and hence $T = \partial E/\partial S$, $-p = \partial E/\partial V$ and $\mu = \partial E/\partial N$. 

25
4 Thermodynamics

4.1.2 Free Energy (Helmholtz Potential)

The potential for a system in the canonical ensemble (with prescribed particle number \( N \), volume \( V \) and temperature \( T \)) is the free energy \( F \). It is connected to the partition function \( Q(N,V,T) \) by

\[
F(N,V,T) = -k_B T \ln Q(N,V,T) .
\]

(4.1.3)

After section 2.2.1 the partition function can be written as the product \( Q = Q_{\text{id}} Q_{\text{ex}} \) and hence the free energy can be written as the sum \( F = F_{\text{id}} + F_{\text{ex}} \). For the classical ideal gas follows from (2.2.6)

\[
F_{\text{id}}(N,V,T) = -N k_B T \left\{ 1 + \ln \left[ \frac{V}{N} \left( \frac{m k_B T}{2 \pi \hbar^2} \right)^{3/2} \right] \right\} .
\]

In thermodynamics the free energy is obtained from the inner energy by the Legendre transformation

\[
F = E - \frac{\partial E}{\partial S} S = E - TS .
\]

Hence the total differential of the free energy is given by

\[
dF = -S dT - p dV + \mu dN + \cdots .
\]

From this can be calculated the entropy \( S = -\partial F/\partial T \), the pressure \( p = -\partial F/\partial V \), the chemical potential \( \mu = \partial F/\partial N \) and the inner energy of the system \( E = \partial (\beta F)/\partial \beta \).

For a system under isothermal-isochoric conditions the free energy becomes a minimum in thermal equilibrium.

4.1.3 Enthalpy

For a system with fixed particle number \( N \), pressure \( p \) and entropy \( S \) the thermodynamical potential is the enthalpy \( H = H(N,p,S) \). It is again obtained by a Legendre transformation of the inner energy

\[
H = E - \frac{\partial E}{\partial V} V = E - pV
\]

with the total differential

\[
dH = T dS + V dp + \mu dN + \cdots .
\]

From this follows \( T = \partial H/\partial S \), \( V = \partial H/\partial p \) and \( \mu = \partial H/\partial N \).

For a system under adiabatic-isobaric conditions the thermal equilibrium is reached in the minimum of the enthalpy.

4.1.4 Free Enthalpy (Gibbs Potential)

The potential for a system in the isothermal-isobaric ensemble (with prescribed particle number \( N \), pressure \( p \) and temperature \( T \)) is the free enthalpy \( G \). It is connected to the partition function \( Q(N,p,T) \) by

\[
G(N,p,T) = -k_B T \ln Q(N,p,T) .
\]

(4.1.4)
As for the canonical ensemble the partition function \( Q(N, p, T) \) can also be written as the product \( Q = Q_{id}Q_{ex} \). For the free enthalpy follows also \( G = G_{id} + G_{ex} \) which results after (2.2.15) in

\[
G_{id}(N, p, T) = -N k_B T \ln \left[ \left( \frac{k_B T}{p} \right)^{5/2} \left( \frac{m}{2\pi \hbar^2} \right)^{3/2} \right].
\]

In thermodynamics the free enthalpy is obtained from the inner energy by two consecutive Legendre transformations

\[
G = E - \frac{\partial E}{\partial S} S - \frac{\partial E}{\partial V} V = E - TS + pV.
\]

Hence the total differential of the free energy is given by

\[
dG = -SdT - Vdp + \mu dN + \cdots.
\]

From this can be calculated the entropy \( S = -\partial G/\partial T \), the volume \( V = \partial G/\partial p \) and the chemical potential \( \mu = \partial G/\partial N \).

For a system under isothermal-isobaric conditions the free enthalpy becomes a minimum in thermal equilibrium.
4 Thermodynamics
Part II

Monte Carlo Simulations of Many Particle Systems
5 Monte Carlo Methods

Die Physik besteht nur aus Näherungen, aber manche Näherungen sind näher als andere.

In part I it was shown that all thermodynamical quantities of a system can be obtained by integration over the phase space, as long as the probability distribution is known. Unfortunately, it isn’t possible to solve these integrals analytically for any “real” system. For a numerical solution, one has to approximate high-dimensional integrals over the phase space. To do this, there are at least two principal possibilities: Monte Carlo (MC) methods and Molecular Dynamics (MD) methods.

The principle of the MC method is as follows: one generates different states of the system according to the probability distribution. For each of these states the interesting observables are calculated and averaged over the states. The idea behind MD methods is to solve the equations of motion of the \( N \) particles numerically. Hence both methods sample the phase space (but in different ways) and approximate the high-dimensional integrals by sums over the sampled states. A comprehensive introduction to classical MC and MD simulations can be found in Frenkel and Smit [29].

The advantage of MC simulations is that one can carry out also “unphysical” moves (in the sense of classical mechanics). The advantage of MD simulations is that all particle coordinates are updated simultaneously. There is also a mixture of these two methods called Hybrid Monte Carlo (see chapter 4.4 in [29]). In this work, only MC simulations are considered.

5.1 Numerical Integration with Random Numbers

There are many different ways to evaluate the one-dimensional integral

\[
I = \int_{a}^{b} f(x) \, dx
\]

numerically. The normal way would be to use a conventional quadrature where the integrand is evaluated at more or less predetermined values of the abscissa. This works well for well-behaved one-dimensional integrals and can—under some circumstances—be extended to low-dimensional integrals (see chapter 4 of Press et al. [70]). Since in simulations of macroscopic systems the integrals over phase space are of very high dimensionality, conventional quadrature isn’t appropriate.
5 Monte Carlo Methods

5.2 Importance Sampling

Let \( x \) be a continuous random variable with a probability density \( w \) (not necessarily normalized). Then the expectation value of a function \( f \) of the random variable is given by

\[
\langle f \rangle = \frac{\int f(x)w(x) \, dx}{\int w(x) \, dx}.
\]

Assuming a positive definite probability density (5.1.1) can be written as

\[
I = \int_a^b f(x)w(x) \, dx = \left( \frac{f}{w} \right) \int_a^b w(x) \, dx.
\]

If \( \int_a^b w(x) \, dx \) can be easily calculated one has a numerical solution of (5.1.1) with an approximation of \( \langle \frac{f}{w} \rangle \). This expectation value can be estimated by the mean value

\[
\langle \frac{f}{w} \rangle \approx \frac{1}{M} \sum_{i=1}^M \frac{f(\xi_i)}{w(\xi_i)},
\]

where the \( \xi_i \) are \( M \) random numbers distributed according to the probability density \( w \). In total, the fundamental theorem of MC integration is (see sections 7.6–7.8 in Press et al. [70])

\[
\int_a^b f(x) \, dx \approx \int_a^b w(x) \, dx \left[ \frac{\bar{T}}{w} \pm \sqrt{\left( \frac{\bar{T}^2}{w^2} - \bar{f}/\bar{w}^2 \right)/M} \right].
\]

While the error of conventional quadrature increases exponentially with the dimension of the integral the error estimate \( \sqrt{\ldots}/M \) in (5.2.4) is independent on the dimension of the integral. This remarkable fact makes MC integration so powerful for integrals of high dimensionality. Since the error goes with \( 1/\sqrt{M} \) one has to quadruplicate \( M \) to double the accuracy.

For numerical evaluations of integrals such as (5.1.1) one has the freedom of making a choice for the probability density \( w \). By choosing \( w \equiv f \) the error would be zero, though nothing would have been gained in this case.

In the naive MC method one uses a random variable distributed uniformly on the interval [\( a, b \)]. In this case the (normalized) probability density is given by \( w(x) = \frac{1}{b-a} \) if \( a \leq x \leq b \), \( 0 \) otherwise and (5.2.2) becomes

\[
\int_a^b f(x) \, dx = (b - a)\langle f \rangle.
\]

\( \langle f \rangle \) is approximated by \( \bar{T} = \frac{1}{M} \sum_{i=1}^M f(\xi_i) \) where the \( \xi_i \) are distributed uniformly on the interval [\( a, b \)]. Since in this case the choice of \( w \) is independent of the function \( f \) calculation time is wasted in regions where \( f \) is approximately zero. That the naive MC method works is shown in figure 5.1 for the integration of the pathological function \( \sin^2 \frac{1}{x} \) (see Newman and Barkema [63]). The MC result is equal within linewidth to the result of numerical integration with state of the art computer algebra systems.
5.3 The Metropolis Monte Carlo Method

The Metropolis Monte Carlo Method

The Metropolis method introduced by Metropolis et al. [59] in 1953 is a method to generate sequences of random numbers (especially random vectors in many dimensions) according to a given probability density $w$. The drawback of this method is that the generated sequences of the states of the system have a sequential correlation. This is no problem in calculating the mean values from above but the error estimate becomes a little bit more complicate. One
method to calculate estimates for the statistical errors are block mean values (see section 6.2.3). Other methods can be found in [63].

Markov Chains

The Metropolis method is based on stationary Markov chains. Let \((x_0, x_1, x_2, \ldots)\) be a sequence of discrete random numbers where the \(x_i\) (the “state” of the system at “time” \(i\)) can take on any values in the set \(\{X_1, X_2, \ldots, X_k\}\) (the possible states of the system). \(P(x_n = X_{i(n)} | x_{n-1} = X_{i(n-1)}, x_{n-2} = X_{i(n-2)}, \ldots, x_0 = X_{i(0)})\) is the conditional probability that \(x_n = X_{i(n)}\) under the condition that \(x_{n-1} = X_{i(n-1)}, x_{n-2} = X_{i(n-2)}, \ldots, x_0 = X_{i(0)}\). Such a sequence of random numbers is called a Markov chain \(\iff P(x_n = X_{i(n)} | x_{n-1} = X_{i(n-1)}, x_{n-2} = X_{i(n-2)}, \ldots, x_0 = X_{i(0)}) = P(x_n = X_{i(n)} | x_{n-1} = X_{i(0)})\). I.e., the conditional probability that \(x\) “at time \(n\)” takes on the value \(X_{i(n)}\) depends only on the value of \(x\) “at time \(n-1\)”.

The \(P(x_n = X_{i(n)} | x_{n-1} = X_{i(n-1)})\) are called transition probabilities. If the transition probabilities \(p_{ij} = P(i \to j) = P(x_n = X_j | x_{n-1} = X_i)\) are independent of the time \(n\) the Markov chain is called stationary (homogenous in time). The \(w_i^{(n)} = P(x_n = X_i)\) are the probabilities for the system to be found in the state \(x_n = X_i\) at time \(n\).

Since the \(p_{ij}\) are probabilities \(0 \leq p_{ij} \leq 1\) \((i, j = 1, \ldots, k)\) and since the system changes from the state \(i\) to any state the \(p_{ij}\) must obey the sum rule

\[
\sum_{j=1}^{k} p_{ij} = 1 \quad (i = 1, \ldots, k).
\]

The Markov matrix \((p_{ij})\) is a so called stochastic matrix.

The idea behind the Metropolis method is as follows. Let the \(w_i\) \((i = 1, \ldots, k)\) be a given probability distribution\(^1\) for the possible states \(\{X_1, X_2, \ldots, X_k\}\). Now one tries to construct the transition probabilities \(p_{ij}\) such that \(w_i^{(n)} = w_i\) “after a long enough time” \((n \to \infty)\). Hence the \(w_i\) are also called equilibrium occupation probabilities \([63]\). Using that \(P(x_n = X_i) = \sum_{j=1}^{k} P(x_n = X_i \cap x_{n-1} = X_j) = \sum_{j=1}^{k} P(x_n = X_i | x_{n-1} = X_j)P(x_{n-1} = X_j)\) (since \(P(A | B) = P(A \cap B) / P(B)\)) one gets \(w_i^{(n)} = \sum_{j=1}^{k} w_j^{(n-1)} p_{ji} = \sum_{j \neq i} w_j^{(n-1)} p_{ji} + w_i^{(n-1)} p_{ii}\). With \(p_{ii} = 1 - \sum_{j \neq i} p_{ij}\) this becomes the master equation\(^2\)

\[
\frac{w_i^{(n)} - w_i^{(n-1)}}{\text{change of the absolute probability for state } i} = \sum_{j \neq i} w_j^{(n-1)} p_{ji} - \sum_{j \neq i} w_i^{(n-1)} p_{ij} \quad (i = 1, \ldots, k).
\]

For the absolute probabilities to be independent of time (equilibrium state) it is necessary that \(0 = w_i^{(n)} - w_i^{(n-1)}\). Additionally, for \(n \to \infty\) one wants to have \(w_i^{(n)} \to w_i\), resulting in

\(^1\)Since a computer can only store a finite number of different numbers with a given precision the difference between a probability distribution and a probability density is of very little importance in computer simulations. Given a probability density \(w = w(x)\) then the set \(\{X_1, X_2, \ldots, X_k\}\) is the set of all storable numbers with a given precision and \(w_i = w(X_i)\) \((i = 1, \ldots, k)\).

\(^2\)The master equation is in fact a set of equations, although it is always called the master equation.
0 = \sum_{j \neq i} w_j p_{ji} - \sum_{j \neq i} w_i p_{ij}. Adding +w_i p_{ii} - w_i p_{ii} one gets the necessary condition for the transition probabilities

\[ 0 = \sum_{j=1}^{k} (w_j p_{ji} - w_i p_{ij}) \quad (i = 1, \ldots, k). \]  

(5.3.3)

With the sum rule (5.3.1) it follows from (5.3.3) that \( \sum_{j=1}^{k} w_j p_{ji} = w_i \), i.e., \( (w_i) \) is a left eigenvector of the matrix \( (p_{ji}) \) with eigenvalue 1. The construction of the transition probabilities is in this sense the inverse of the eigenvalue problem: given an eigenvector with an eigenvalue find one associated matrix.

**Detailed Balance**

There are many ways to fulfill the necessary condition for the transition probabilities in (5.3.3). To avoid a dynamic equilibrium with a limit cycle one imposes the condition of detailed balance [63] (see also figure 3.6 in [29])

\[ w_j p_{ji} = w_i p_{ij}, \]  

(5.3.4)

i.e., every term of the sum in (5.3.3) vanishes. A Markov chain with this property is for obvious reasons called reversible. Since the probability distribution \( w_i \) \( (i = 1, \ldots, k) \) is given one gets the relations

\[ p_{ij} = \frac{w_j}{w_i} \]  

for the transition probabilities. The remarkable fact about (5.3.5) is that only the ratios \( w_j/w_i \) are important. I.e., the \( w_i \) \( (i = 1, \ldots, k) \) need not be normalized since the normalization constant \( (\sum_{i=1}^{k} w_i)^{-1} \) cancels out. Choosing \( p_{ij} = w_j/\sum_{i=1}^{k} w_i \) independent of \( i \) would fulfill (5.3.5) but nothing would have been gained in this case because then the probability distribution \( w_i \) could have been sampled directly.

**The Metropolis Method**

To sum up, one has a given probability distribution \( w_i \) \( (i = 1, \ldots, k) \) for the possible states \( \{X_1, \ldots, X_k\} \) of a system. If the system is in the state \( x_{n-1} = X_i \) at time \( n - 1 \) one tries to construct transition probabilities \( p_{ij} \) for the system to be found in the state \( x_n = X_j \) at time \( n \) such that the probability \( P(x_n = X_j) \) for the system to be in the state \( x_n = X_j \) at time \( n \) tends to \( w_j \) for large \( n \) (\( P(x_n = X_j) \to w_j \) for \( n \to \infty \)). These transition probabilities are subject to the following conditions:

- the \( p_{ij} \) are time-independent,
- \( 0 \leq p_{ij} \leq 1 \),
- \( \sum_{j=1}^{k} p_{ij} = 1 \) for \( i = 1, \ldots, k \),
- \( \frac{p_{ij}}{p_{ji}} = \frac{w_j}{w_i} \) and
starting from some state it is possible to reach any other state of the system in a finite number of time steps (ergodicity).

These constraints still leave a good deal of freedom for the construction of the transition probabilities \( p_{ij} \). The Metropolis method proposed by Metropolis et al. [59] consists of splitting the transition probabilities into a product of a probability \( t_{ij} = T(i \to j) \) (called selection probability in [63]) for a trial move from state \( i \) to state \( j \) and a probability \( a_{ij} = A(i \to j) \) for the acceptance of this trial move:

\[
p_{ij} = t_{ij} a_{ij}.
\] (5.3.6)

What has been gained by this splitting? For \( i = j \), i.e., transitions \( i \to i \), the condition of detailed balance for \( p_{ii} \) is always satisfied, independent of its actual value. This gives some flexibility about how to choose the other transition probabilities with \( i \neq j \): An adjustment of the value of any of the \( p_{ij} \) can be compensated by an equal but opposite adjustment of \( p_{ii} \) (with the constraint \( 0 \leq p_{ii} \leq 1 \)) such that the sum rule (5.3.1) is obeyed. If with \( p_{ij} \) the according \( p_{ji} \) is also adjusted such that the ratio of the two is preserved the condition of detailed balance (5.3.5) remains satisfied. [63]

Now given an algorithm that generates the state \( j \) from the current state \( i \) the new state \( j \) should be accepted with a probability \( a_{ij} \). With probability \( 1 - a_{ij} \) the system should stay in state \( i \). The acceptance ratios can be any number between zero and one. This gives complete freedom about how to choose the selection probabilities \( t_{ij} \), since the condition of detailed balance in (5.3.5) only fixes the ratio

\[
\frac{p_{ij}}{p_{ji}} = \frac{t_{ij} a_{ij}}{t_{ji} a_{ji}}.
\]

The ratio \( a_{ij}/a_{ji} \) can take on any value between zero and infinity, which means that both \( t_{ij} \) and \( t_{ji} \) can take any values one likes to satisfy (5.3.5). [63]

For the Metropolis method one assumes that the trial moves \( i \to j \) and \( j \to i \) have equal probability, i.e., \( t_{ij} = t_{ji} \). Normally, one chooses the new state \( j \) in the "neighborhood of the state \( i \). For the acceptance probabilities one has again \( a_{ij}/a_{ji} = w_j/w_i \).

In the original Metropolis method (asymmetric rule) the trial move \( i \to j \) will be accepted in any case if \( w_j \geq w_i \), otherwise it will be accepted only with a probability of \( w_j/w_i \). Hence the acceptance probabilities are given by

\[
a_{ij} = \min(1, \frac{w_j}{w_i}).
\] (5.3.7)

It is trivial to show that from this follows \( a_{ij}/a_{ji} = w_j/w_i \). The asymmetric rule is used for all the simulations in this work.

With the symmetric rule the acceptance probabilities for the trial moves \( i \to j \) are given by

\[
a_{ij} = \frac{w_j}{w_i + w_j}.
\] (5.3.8)

Again \( a_{ij}/a_{ji} = w_j/w_i \) is satisfied. The symmetric rule is often used for the simulation of spin systems.
What happens with the rejected trial moves? Since every trial move $t_{ij}$ moves the system from the state $i$ to any other state $j$, $\sum_j t_{ij} = 1$ and it follows $p_{ii} = 1 - \sum_{j \neq i} p_{ij} = \sum_j t_{ij}$, $t_{ii}$ is the probability that the system stays in the state $i$, $\sum_{j \neq i} t_{ij} (1 - a_{ij})$ is the probability that the trial move leads the system to state $j$ and is rejected. Hence transitions $i \rightarrow i$ happen when the new generated state is the old state or when a trial move is rejected. I.e., rejected moves have to be counted as transitions $i \rightarrow i$ (see also figure 3.7 in [29]).

Figure 5.2 shows an algorithm for the original Metropolis method (asymmetric rule).

<table>
<thead>
<tr>
<th>To generate a sample $(x_0, x_1, \ldots, x_n, x_{n+1}, \ldots)$ of a random variable $x$ with a probability density $w(x)$ one starts with an arbitrary initial state $x_0$ and generates $x_{n+1}$ successively from $x_n$ by the following steps:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Trial move: Choose a $x'$ in the “neighborhood” of $x_n$. E.g., $x' = x_n + (\eta - \frac{1}{2})\Delta$ where $\eta$ is uniformly distributed in $[0, 1)$ (this satisfies $T(i \rightarrow j) = T(j \rightarrow i)$). As a rule of thumb, the maximum displacement $\Delta$ is chosen such that about 50% of all trial moves are accepted (but see also section 3.3 in [29]).</td>
</tr>
<tr>
<td>2. Acceptance test: If $w(x') \geq w(x_n)$ then $x_{n+1} = x'$ is accepted with probability 1. Otherwise generate another random variable $\eta'$ distributed uniformly on $[0, 1)$. If $\eta' \leq w(x')/w(x_n)$ then $x_{n+1} = x'$ else $x_{n+1} = x_n$. I.e., the move is accepted with probability $A(n \rightarrow n + 1) = \min(1, w(x')/w(x_n))$.</td>
</tr>
</tbody>
</table>

Figure 5.2: Algorithm for the Metropolis method (asymmetric rule)

A method of calculating the transition probabilities $P(i \rightarrow j)$ differently than Metropolis et al. is continuous time MC, the BKL algorithm of Bortz, Kalos and Lebowitz (1975). It is especially suitable for systems at low temperatures. An introduction to it can be found in the book by Newman and Barkema [63]. In the present work only the Metropolis method is considered.
6 Classical Monte Carlo Simulations

Those who can, do; those who can’t, simulate.

6.1 Lennard-Jones Potential

One essential point in MC simulations (and in computer simulations of many-particle systems in general) is the “choice” of the interaction between two molecules [17]. For spherically symmetric molecules one assumes a simple model using a spherical interaction that depends only on the distance between two molecules.\(^1\) For more complex molecules one has to introduce different levels of anisotropy in the model potential.

The simplest spherical model are hard spheres. In this model the molecules don’t attract each other. In every collision between two molecules momenta and energy are exchanged elastically. I.e., the intermolecular potential is zero as long as the distance \( r_{ij} \) between the molecules is greater than their diameter \( \sigma \); for distances lower than the diameter the potential is infinite: \( U_{ij}(r_{ij}) = 0 \text{ for } r_{ij} \geq \sigma \text{ and } U_{ij}(r_{ij}) = \infty \text{ for } r_{ij} < \sigma \). Since in this model there are no attractive forces one has assumed that a fluid of hard spheres cannot undergo a phase transition. Hoover and Ree [41] proved the existence of a melting transition in such a system with MC simulations.

The Lennard-Jones potential is a semi-empiric potential for the intermolecular interaction of two molecules that depends only on their mutual distance. Let \( r_{ij} := |r_i - r_j| \) denote the distance between the particles \( i \) and \( j \). Then the two-particle interaction potential \( U_{ij} \) from section 2.2 is given by (see figure 6.1)

\[
U_{ij}(r_{ij}) = 4\varepsilon \left[ \left( \frac{\sigma}{r_{ij}} \right)^{12} - \left( \frac{\sigma}{r_{ij}} \right)^{6} \right].
\] (6.1.1)

The potential energy vanishes at \( r_{ij} = \sigma \) and for \( r_{ij} \to \infty \). The minimal potential energy \( -\varepsilon \) is reached at \( r_0 = \sqrt[6]{2}\sigma = 1.12462\ldots \sigma \).\(^2\)

Two spherical molecules with vanishing electrical moments of all orders\(^3\) attract each other due to van der Waals forces caused by induced dipoles. This gives the \( r^{-6} \) dependence of the interaction potential (see Kittel [50]). As long as the molecules are distant enough they feel only attractive forces. When two molecules approach each other the electron clouds begin to overlap. For small enough distances the Pauli exclusion principle gives an additional repulsive contribution to the forces. This repulsive force is modeled by the \( r^{-12} \) dependence of the

---

\(^{1}\)In fact, the simplest model would be no interaction at all. But normally, one doesn’t need to simulate an ideal gas on a computer.

\(^{2}\)\(\sigma\) is called Lennard-Jones diameter and \( r_0 \) is called equilibrium distance. Another formulation of (6.1.1) in terms of \( r_0 \) instead of \( \sigma \) would be \( U_{ij}(r_{ij}) = \varepsilon \left( \frac{r_0}{r_{ij}} \right)^{12} - 2\left( \frac{r_0}{r_{ij}} \right)^{6} \).

\(^{3}\)I.e., the time averages of the electrical moments vanish.
6.2 Canonical Ensemble

While for the evaluation of integrals such as (5.1.1) one has the freedom of choosing a probability density the probability density for the evaluation of expectation values of observables is already determined by statistical mechanics (see chapter 2). The normalized probability density in the canonical ensemble is given by

$$w(r^N, p^N) = \frac{1}{Z} e^{-\beta H(r^N, p^N)} \text{ (see section 2.2.1).}$$

Most relevant observables depend only on the coordinates of the molecules. In this case the normalized probability density reduces to

$$w(r^N) = \frac{1}{Z} e^{-\beta U(r^N)}.$$  \hspace{1cm} (6.2.1)

Since $w(r^N)$ is normalized the expectation value of the observable $O$ is given by

$$\langle O \rangle = \int_V w(r^N) O(r^N) \, dr^N.$$  \hspace{1cm} (6.2.2)

The evaluation of the configurational integral $Z$ from (2.2.7) is problematic. Fortunately, one doesn’t need the actual value of $Z$ since the probability density need not be normalized for the MC evaluation of $\langle O \rangle$. 

Lennard-Jones potential because it is easy to evaluate and gives good results for a variety of systems (especially noble gases). 

Figure 6.1: The Lennard-Jones potential from equation (6.1.1) (solid curve). Also shown are the repulsive (dashed curve) and the attractive part (dash-dotted curve).
The initial configuration of the coordinates is arbitrary. Normally, one starts with a random initial configuration (see section C.1.1) or with an (fcc) lattice as the initial configuration (see section C.1.2). Before the actual simulation starts this initial configuration is equilibrated.

1. The system being in state \( n \) one performs the following steps to transfer the system to the state \( n+1 \):
   a) Trial move: Select a particle \( i \) randomly from the \( N \) total particles and calculate the potential energy of the current state \( U(r_N^N) \). Move the particle \( i \) to the new position \( r'_i = r_i + (\eta - \frac{1}{2})\Delta \) where the vector \( \eta \) is uniformly distributed in \([0,1]^3\). \( \Delta \) is chosen such that about 50\% of the trial moves are accepted. Calculate the potential energy of this intermediate state \( U(r'_N) \).
   b) Acceptance test: Accept the particle move with the probability
      \[
      \mathcal{A}_{pm}(n \rightarrow n + 1) = \min \left( 1, e^{-\beta[U(r'_N) - U(r_N^N)]} \right). 
      \]
      If the particle move is accepted then \( r_{n+1}^N = r'_N \) else \( r_{n+1}^N = r_n^N \).

2. Skipping: To avoid costly evaluations of observables in statistically similar states the steps a and b are repeated \( N_{n\text{skip}} \) times (\( n_{\text{skip}} \) passes) before calculating the values of the observables and averaging them.

3. Continue with step 1.

Figure 6.2: Algorithm for MC simulations in the canonical ensemble

### 6.2.1 Particle Moves

Let \( r_i^N \) be the coordinates of the particles in state \( i \). Provided that the coordinates can be sampled from the probability density \( w(r_N) \) the expectation value \( \langle O \rangle \) can be approximated by the sample mean value

\[
\overline{O} = \frac{1}{M} \sum_{i=1}^{M} O(r_i^N), \tag{6.2.3}
\]

where \( O(r_i^N) \) is the value of the observable in state \( i \). The evaluation of \( \langle O \rangle \) reduces therefore to a sampling of the states \( r_i^N \) according to the probability density \( w(r_N) \) as described in section 5.3.

According to the algorithm of the Metropolis method from figure 5.2 one has to start with some arbitrary initial configuration of the coordinates of the particles. Then one has to perform trial moves to another configuration of the coordinates of the particles. In simulations in the canonical ensemble these trial moves are single particle moves.

The algorithm is as follows (see figure 6.2): One starts with an arbitrary initial configuration of the coordinates. Then one performs trial moves to an intermediate configuration. If the
potential energy is smaller than the potential energy in the current state then the trial move is accepted. If the potential energy of the trial state is greater then the move is accepted with probability \( \exp(-\beta \Delta U) \), where \( \Delta U \) is the difference of the potential energies. For the calculation of \( \Delta U \) one only needs to consider interactions with the moved particle. Hence this is an order \( O(N) \)-process. \( N \) attempted particle moves are called one “pass”. I.e., on the average every particle has undergone a trial move in every pass.

The so generated states \( n \) and \( n + 1 \) are very similar. To avoid costly evaluations of observables in very similar states one generates \( Nn_{\text{skip}} \) configurations (\( n_{\text{skip}} \) passes) before averaging the observables. The expression \( 10 \times 10000 \) passes means that \( 10 \times 10000 \times N \) configurations have been generated and for 10000 of these the observables have been averaged.

### 6.2.2 Particle Swaps in Mixtures

In simulations of mixtures of two substances it can take a considerable amount of time till the substances get mixed only by single particle moves. For such simulations another type of trial move is introduced: the swapping of particles. These moves are somewhat similar to moving two particles at the same time. One chooses randomly a particle \( i_1 \) from the \( N_1 \) particles of substance 1 and another random particle \( i_2 \) from the \( N_2 \) particles of substance 2. The new coordinates of particle \( i_1 \) are the old coordinates of particle \( i_2 \) and vice versa. The rest of this kind of trial move is similar to a normal particle move. Normally, every tenth particle move is replaced by a particle swap, but this is an arbitrary choice.

### 6.2.3 Block Mean Values

To obtain an estimate of the statistical error of the simulation results one usually splits the simulations in blocks of equal length (of the order of 10 blocks are normally used). The averages are then calculated for each block separately and the final result is the mean of these block mean values. Provided that the blocks are statistically independent, the statistical error of this final mean value is given by the standard deviation of the block mean values divided by \( \sqrt{n_{\text{blocks}}} \), where \( n_{\text{blocks}} \) is the number of simulated blocks.

That neither the accumulated average \( a_l \) after \( l \) blocks nor the block mean values are smooth functions is shown in figure 6.3. Calculating the block mean values from the accumulated averages is some kind of differentiation: The accumulated average after \( l \) blocks is given by

\[
a_l = \frac{1}{lM} \sum_{i=1}^{lM} x_i,
\]

where \( M \) is the number of passes per block. After the first block the accumulated average is equal to the block mean value. The block mean value of block \( l \) is given by \( (lM a_l - (l - 1) M a_{l-1}) / M = l a_l - (l - 1) a_{l-1} \). Hence the block mean values are calculated by the following recurrence relation

\[
\begin{align*}
\overline{x}_1 &= a_1, \\
\overline{x}_l &= l a_l - (l - 1) a_{l-1},
\end{align*}
\]

(6.2.4)
6.2 Canonical Ensemble

Figure 6.3: Accumulated average and block mean values for the density of $D_2$ at $p = 1.5$ atm and $T = 20.4$ K. The simulation consisted of $N = 108$ classical particles and $P = 64$ beads per particle.

6.2.4 Computational details

In this section several details for MC simulations in general will be discussed. An extended discussion of these details can be found in section 3.2.2 in [29].

Periodic Boundary Conditions and Minimum Image Convention

MC simulations of atomic or molecular systems aim to provide information about the properties of a macroscopic sample. Yet, it isn’t possible to simulate system with more than a few hundred thousand particles. Clearly, this number is far from the thermodynamic limit ($N \sim 10^{23}$). Additionally, most of the particles are near the surface of the simulation box. As Frenkel and Smit [29] state, in a simple cubic crystal of 1000 atoms, about 49% of all atoms are at the surface, and for $10^6$ atoms this fraction has decreased to only 6%. Hence finite-size effects influence the simulation results. One simple way to reduce these effects is to assume that the cubical simulation box is surrounded by (an infinite number) of copies of itself. The distance between two particles has then to be taken modulo the length of the simulation box (minimum image convention).
Cutoff and Tail Corrections

To calculate the total potential energy of the system one has to evaluate $N(N-1)/2$ summands. Hence this is an $N^2$ operation. Especially when the interaction potential is hard to evaluate, calculating the total potential energy can be very time consuming. If the total potential energy of a given particle $i$ is dominated by interactions with neighboring particles that are closer than some cutoff distance $r_c$ the intermolecular interaction is called short-ranged. In this case one can consider only interactions of particles closer than the cutoff distance and average the interactions with particles that are farther away into an additional constant. I.e., the potential energy of the system can be approximately calculated by

$$U \approx \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} U_{ij}^c(r_{ij}) + U_{\text{tail}}$$

where

$$U_{ij}^c(r_{ij}) = \begin{cases} U_{ij}(r_{ij}) & \text{if } r_{ij} < r_c, \\ 0 & \text{otherwise} \end{cases}$$

and $r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$. The tail correction $U_{\text{tail}}$ is given by

$$U_{\text{tail}} = \frac{N\rho}{2} \int_{r_c}^{\infty} U_{ij}(r) 4\pi r^2 \, dr.$$ 

In the case of a Lennard-Jones potential of the form (6.1.1) the tail correction becomes

$$U_{\text{tail}} = \frac{8N}{3} \pi \rho \sigma^3 \left[ \frac{1}{3} \left( \frac{\sigma}{r_c} \right)^9 - \left( \frac{\sigma}{r_c} \right)^3 \right].$$

In the simulations in the present work the cutoff distance was chosen to be $r_c = \frac{1}{2} V^{1/3}$. In the isothermal-isobaric ensemble the cutoff and the tail corrections have to be recalculated after every volume move.

Reduced Units

For simulations at the microscopic level the standard SI units aren’t very suitable. One prefers units that are of the order of the calculated quantities. The Lennard-Jones potential (see section 6.1) has two parameters: the Lennard-Jones diameter $\sigma$ and the energy $\varepsilon$. In simulations of Lennard-Jones systems it is useful to use $\sigma$ as the unit of length and $\varepsilon$ as the unit of energy. All other properties can then be also expressed in terms of these Lennard-Jones units. Another useful system of units for simulations of many particle systems are physico-chemical units. Table 6.1 compares these two unit systems and lists some conversions between these and some SI units.

From table 6.1 follows for $\beta = 1/k_B T$

$$\beta_{\text{LJ}} = \varepsilon \beta_{[1/(\text{J})]} = \frac{N_A \varepsilon}{10^3 N} \beta_{([\text{kJ/mol}])^{-1}}$$

and for $q = \hbar^2/2m$

$$q_{\text{LJ}} = \frac{1}{\varepsilon \sigma^2} q_{[\text{Jm}^2]} = \frac{N}{10^{12} N_A \varepsilon \sigma^2} q_{[\text{kJ/mol nm}^2]}.$$
Table 6.1: Reduced units used in the simulations: Lennard-Jones (LJ) and physico-chemical (PC) units. The last column gives some conversions between these and some SI units. The usual mass density is given by 

\[ \rho_n = \frac{1}{\sigma^3}, \quad \rho_L = \frac{1}{\sigma^3}, \quad \rho_P = \frac{1}{\sigma^3}, \quad \rho_T = \frac{1}{\sigma^3}. \]

<table>
<thead>
<tr>
<th>Quantity</th>
<th>LJ Unit</th>
<th>PC Unit</th>
<th>Conversions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length</td>
<td>( \sigma )</td>
<td>1 nm</td>
<td>( r_{LJ} = \frac{1}{\sigma} r_{[nm]} )</td>
</tr>
<tr>
<td>Volume</td>
<td>( \sigma^3 )</td>
<td>1 nm(^3)</td>
<td>( V_{LJ} = \frac{1}{\sigma^3} V_{[nm]^3} )</td>
</tr>
<tr>
<td>Density</td>
<td>( 1/\sigma^3 )</td>
<td>1 nm(^3)</td>
<td>( \rho_{LJ} = \frac{1}{\sigma^3} \rho_{[nm]^{-3}} = 10^{27} \sigma^3 \rho_{[nm]^{-3}} = 10^3 N_A \sigma^3 \rho_{[mol]^{-1}} )</td>
</tr>
<tr>
<td>Energy</td>
<td>( \varepsilon )</td>
<td>1 kJ/mol</td>
<td>( E_{LJ} = \frac{1}{\varepsilon} E_{[J]} = \frac{N}{N_A \varepsilon} E_{[J/mol]} = 10^3 N_A \varepsilon E_{[kJ/mol]} )</td>
</tr>
<tr>
<td>Pressure</td>
<td>( \varepsilon/\sigma^3 )</td>
<td>1 atm</td>
<td>( p_{LJ} = \frac{1}{\varepsilon} p_{[atm]} = 10^3 N_A \varepsilon p_{[mol]^{-1}} )</td>
</tr>
<tr>
<td>Temperature</td>
<td>( \varepsilon/k_B )</td>
<td>1 K</td>
<td>( T_{LJ} = \frac{k_B}{\varepsilon} T_{[K]} )</td>
</tr>
</tbody>
</table>

6.3 Isothermal-Isobaric Ensemble

According to section 2.2.2 the normalized probability density in the isothermal-isobaric ensemble is given by 

\[ w(V, r^N, p^N) = \left(\frac{1}{Q}\right) e^{-\beta[H(r^N, p^N) + pV]}, \]

For observables depending only on the coordinates this probability density reduces to

\[ w(V, s^N) = \frac{V^N}{Z} e^{-\beta[U(V^{1/3}, s^N) + pV]} = \frac{1}{Z} e^{-\beta\left[U(V^{1/3}, s^N) + pV - \frac{N}{\sigma^3} \ln V\right]}, \]

where the scaled coordinates from (2.2.13) have been used.

6.3.1 Particle Moves

The full phase space of the isothermal-isobaric ensemble has \( 6N + 1 \) dimensions. The \( 6N \)-dimensional coordinate subspace can be sampled using conventional particle moves as trial moves. These particle moves are identical to the ones in the canonical ensemble (see section 6.2). Since for particle moves the volume remains constant a particle move is accepted with a probability

\[ A_{pm}(n \to n + 1) = \min\left(1, e^{-\beta \Delta U}\right), \]

where \( \Delta U \) is the difference in the potential energy between system at “time” \( n \) and the system in the trial state \( \prime \). Again the calculation of \( \Delta U \) is of order \( O(N) \).

6.3.2 Volume Moves

The remaining 1-dimensional subspace of the phase space is sampled with “volume moves” as trial moves, i.e., the size of the cubical simulation box is changed from \( V_n \) to \( V' = V_n + \)
\((\eta - \frac{1}{2})\Delta V\). This means that the coordinates of the \(N\) particles have also to be changed from \(r^N_n = V^{1/3}N s^N_n\) to \(r'^N_n = V'^{1/3}N s'^N_n\). Such a volume move is accepted with the probability
\[
A_{vm}(n \rightarrow n + 1) = \min \left( 1, e^{-\beta \left[ \Delta U + p(V' - V_n) - \frac{N}{3} \ln \frac{V'}{V_n} \right]} \right).
\]
Since now the distances between all the particles are changed the potential energy of the system has to be recalculated completely. Because this recalculation of the potential energy is an \(O(N^2)\) process one makes volume moves not too frequently. A rule of thumb says that for \(N\) particle moves one should make only one volume move. A pass now consists of \(N\) attempted particle moves and 1 attempted volume move.

Instead of attempting random changes in the volume itself, it is often better to perform trial moves in the logarithm of the volume. The partition function from (2.2.14) then becomes
\[
Q(N, p, T) = \frac{\beta_p}{N!A^{3N}} \int_0^\infty V^{N+1} e^{-\beta p V} \int_\mathbb{R}^N e^{-\beta U(V^{1/3} s^N)} ds^N dV
\]
with the probability density
\[
w(V, s^N) = \frac{V^{N+1}}{Z} e^{-\beta U(V^{1/3} s^N) + pV} = \frac{1}{Z} e^{-\beta \left[ U(V^{1/3} s^N) + pV - \frac{N+1}{3} \ln V \right]}.
\]
\[
(6.3.2)
\]
For particle moves nothing is changed but the acceptance probability for volume moves becomes
\[
A_{vm}(n \rightarrow n + 1) = \min \left( 1, e^{-\beta \left[ \Delta U + p(V' - V_n) - \frac{N+1}{3} \ln \frac{V'}{V_n} \right]} \right).
\]
\[
(6.3.3)
\]
**Volume Moves and Lennard-Jones potential**

As has already been mentioned the recalculation of the potential energy is an \(O(N^2)\) process. If the pair-potential is of Lennard-Jones type (6.1.1) the repulsive part is simply scaled by \((V/V')^4\) and the attractive part is scaled by \((V/V')^4\). For more realistic potentials the recalculation of the potential energy is not so simple.

The expectation value \(\langle O \rangle\) of the observable \(O(r^N)\)
\[
\langle O \rangle = \int_0^\infty \int_\mathbb{R}^N w(V, s^N) O(V^{1/3} s^N) ds^N dV
\]
\[
(6.3.4)
\]
is again approximated by the mean value
\[
\overline{O} = \frac{1}{M} \sum_{i=1}^M O(V^{1/3}_i s^N_i),
\]
where \(V_i\) is the volume of the system and \(s^N_i\) are the scaled coordinates of the particles in configuration \(i\).

The computational details are the same as for the canonical ensemble in section 6.2.4. The complete algorithm for MC simulations in the isothermal-isobaric ensemble is shown in figure 6.4.
The initial configuration consists of a starting volume and an arbitrary choice of the coordinates. The starting volume has to be guessed from the given pressure and temperature or it can be taken from previous runs of similar systems. For the starting configuration of the coordinates one can again use random coordinates (see section C.1.1) or an (fcc) lattice (see section C.1.2). Additionally, one has to choose a probability $p_{vm}$ for a volume move (normally, $p_{vm} \approx 1/N$).

Before the actual simulation starts this initial configuration is equilibrated. To do this the volume is held constant at first and only particle moves are carried out. After some time also volume moves are permitted.

1. The system being in state $n$ one performs the following steps to transfer the system to the state $n+1$:

   a) Trial move: With probability $1 - p_{vm}$ perform a particle move: Select a particle $i$ randomly from the $N$ total particles and calculate the potential energy of the current state $U(r^N_n)$. Move the particle $i$ to the new position $r'_i = r_i + (\eta - \frac{1}{2}) \Delta_{pm}$ where the vector $\eta$ is uniformly distributed in $[0, 1)^3$. $\Delta_{pm}$ is chosen such that about 50% of the particle moves are accepted. Calculate the potential energy of this intermediate state $U(r'_N)$.

   b) Acceptance test: Accept the particle move with the probability

   $$A_{pm}(n \rightarrow n+1) = \min\left(1, e^{-\beta[U(r'_N) - U(r^N_n)]}\right).$$

   If the particle move is accepted then $r^N_{n+1} = r'_N$ else $r^N_{n+1} = r^N_n$.

   Accept the volume move with the probability

   $$A_{vm} = \min\left(1, e^{-\beta[U(V^N_{1/3}s^N_n) - U(V^N_{1/3}s^N_n)]} + p(V_{n+1} - V_n)\frac{N+1}{N}\ln\frac{V_{n+1}}{V_n}\right),$$

   If the volume move is accepted then $r^N_{n+1} = V^{1/3}s^N_{n+1}$ else $r^N_{n+1} = V^{1/3}s^N_n$.

   Of course, for a volume move $s^N_{n+1} = s^N_n$.

2. Skipping: As in the canonical ensemble the steps a and b are repeated $Nn_{skip}$ times ($n_{skip}$ passes) before calculating the values of the observables and averaging them.

3. Continue with step 1.

Figure 6.4: Algorithm for MC simulations in the isothermal-isobaric ensemble
6.3.3 Particle Swaps in Mixtures

For mixtures additional trial moves are particle swaps. Their implementation in the isothermal-isobaric ensemble is identical to their implementation in the canonical ensemble (see section 6.2.2).
7 Path-Integral Monte Carlo Simulations

In the following only a short overview is given on how to extend a classical MC simulation program into a Path-Integral Monte Carlo program. The details can be found in the program code in section C.2.

7.1 Canonical Ensemble

Due to the classical isomorphism (see section 3.2.3) Path-Integral Monte Carlo simulations are very similar to classical MC simulations. The only big difference is an additional averaging over the beads of the ring-polymers.

7.1.1 Particle Moves

Single particle moves now consist of moving the center of mass of one ring-polymer as in the classical case (see section 6.2) and the generation of the relative distances of the beads to the center of mass. This generation of the bead-coordinates is described in section 7.3.\(^1\)

7.1.2 Particle Swaps in Mixtures

As for the classical mixtures it is useful to exchange the positions of two quantum particles of different substances. But in the quantum mechanical case only the centers of mass of the classical ring-polymers have to be exchanged. The relative distances of the beads to the centers of mass have to be kept constant.

7.2 Isothermal-Isobaric Ensemble

7.2.1 Particle Moves

The single particle moves are as in the canonical ensemble.

\(^1\)A simpler method is described by Vesely [91]: In addition to choosing a maximum displacement for the ring-polymers one chooses a smaller maximum displacement for the beads. A single particle move then consist of moving the center of mass and all beads according to their maximum displacements. The “spring” energy has then to be included in the acceptance probability. This method results in longer simulation runs since the properties of the system converge more slowly. Nevertheless, for dense systems this method can be better than the Fourier method from section 7.3.
7.2.2 Volume Moves

The volume moves are as in the classical case of section 6.3. If the volume is changed the centers of mass have to be changed accordingly. Additionally, the “size” of the ring-polymers has also to be changed as the cube root of the volume ratios. (This is contrary to the advice by Frenkel and Smit [29] on “real” polymers. Their size has to remain unchanged in volume moves.) In the acceptance probability the change in the spring energy has to be included. Also the term \((N + 1)/\beta \ln V'/V_n\) has to be changed to \((NP + 1)/\beta \ln V'/V_n\) (when performing moves in \(\ln V\)).

7.2.3 Particle Swaps in Mixtures

For the particle swaps holds the same as in the canonical ensemble.

7.3 Generation of the Bead-Coordinates

As has been shown in chapter 3 the distribution of the bead coordinates is proportional to the multi-variate Gauß-distribution

\[
w(\mathbf{r}^{(0)}, \ldots, \mathbf{r}^{(P-1)}) \propto \exp \left[ -\frac{mP}{2\beta\hbar^2} \sum_{k=0}^{P-1} (\mathbf{r}^{(k+1)} - \mathbf{r}^{(k)})^2 \right],
\]

(7.3.1)

with \(\mathbf{r}^{(P)} \equiv \mathbf{r}^{(0)}\). (7.3.1) can be split into

\[
w(\mathbf{r}^{(0)}, \ldots, \mathbf{r}^{(P-1)}) = w_x(x^{(0)}, \ldots, x^{(P-1)}) w_y(y^{(0)}, \ldots, y^{(P-1)}) w_z(z^{(0)}, \ldots, z^{(P-1)}),
\]

where for the \(x\)-coordinates (and analogous for the \(y\)- and \(z\)-coordinates) holds

\[
w_x(x^{(0)}, \ldots, x^{(P-1)}) \propto \exp \left[ -\frac{mP}{2\beta\hbar^2} \sum_{k=0}^{P-1} (x^{(k+1)} - x^{(k)})^2 \right],
\]

(7.3.2)

with \(x^{(P)} \equiv x^{(0)}\).

A general way to produce multi-variate Gauß-distributed random numbers is explained in section 3.2.5 in Vesely [91]. For the special case in (7.3.2), a discrete Fourier transform (DFT) method can be used. The drawback of this simple method is that \(P\) has to be a power of 2. Defining

\[
S := \alpha \sum_{k=0}^{P-1} (x^{(k+1)} - x^{(k)})^2,
\]

inserting the DFT\(^2\)

\[
x^{(k)} = \sum_{l=0}^{P-1} \tilde{x}_l e^{2\pi i kl/P} \quad (k = 0, \ldots, P - 1)
\]

(7.3.3)

\(^2\)(7.3.3) obviously fulfills the desired relation \(x^{(P)} \equiv x^{(0)}\).
and using
\[
\sum_{k=0}^{P-1} e^{2\pi i k (l+l')/P} = P \delta_{l,-l'}, \tag{7.3.4}
\]
one gets
\[
S = 2 \alpha P \sum_{l=0}^{P-1} |\tilde{x}_l|^2 \left[ 1 - \cos \frac{2\pi l}{P} \right].
\]

From \(x^{(k)} \in \mathbb{R} (k = 0, \ldots, P-1)\) follows \(\tilde{x}_l^* = \tilde{x}_{P-l} (l = 0, \ldots, P-1)\), i.e., \(c_l = c_{P-l}\) and \(d_l = -d_{P-l} (\Rightarrow d_0 = d_{P/2} = 0)\). Hence \(S\) becomes
\[
S = 4 \alpha \left\{ (c_1^2 + d_1^2) \left[ 1 - \cos \frac{2\pi}{P} \right] + (c_2^2 + d_2^2) \left[ 1 - \cos \frac{2\pi}{P} \right] + \ldots + c_{P/2}^2 \right\}
\]
and it follows
\[
e^{-S} = e^{-4 \alpha c_1^2 \left[ 1 - \cos \frac{2\pi}{P} \right]} e^{-4 \alpha d_1^2 \left[ 1 - \cos \frac{2\pi}{P} \right]} e^{-4 \alpha c_2^2 \left[ 1 - \cos \frac{2\pi}{P} \right]} e^{-4 \alpha d_2^2 \left[ 1 - \cos \frac{2\pi}{P} \right]} \cdots e^{-4 \alpha c_{P/2}^2}.
\]

Hence to generate \(x^{(0)}, \ldots, x^{(P-1)}\) with probability density (7.3.2) one has to perform the following steps:

1. Draw the \(c_1, d_1, \ldots, c_{P/2}\) from normal distributions with variances
\[
\sigma_{c_l}^2 = \sigma_{d_l}^2 = \begin{cases} \frac{1}{8 \alpha P} & (1 \leq l < P/2), \\ \frac{\beta \hbar^2}{4 m P^2} & (l = P/2). \end{cases}
\]

2. Calculate the \(x^{(k)}\) by (7.3.3)
\[
x^{(k)} = c_0 + 2 c_1 \cos \frac{2\pi k}{P} - 2 d_1 \sin \frac{2\pi k}{P} + 2 c_2 \cos \frac{2\pi k}{P} - 2 d_2 \sin \frac{2\pi k}{P} + \cdots + (-1)^k c_{P/2}.
\]

\(c_0\) is the center of mass of the beads. Since the polymer is always moved to the center of mass of the classical particle, one can set \(c_0 = 0\).

In Lennard-Jones units the variances become
\[
\sigma_{c_l}^2 = \sigma_{d_l}^2 = \begin{cases} \frac{q_{LJ}}{2 P^2 T_{LJ} [1 - \cos \frac{2\pi}{P}]^2} & (1 \leq l < P/2), \\ \frac{q_{LJ}}{2 P^2 T_{LJ}} & (l = P/2). \end{cases}
\]

---

\(^3\)The case \(l = -l'\) is clear. Now the case \(l \neq -l'\): Let \(z := e^{2\pi i (l+l')/P}\). Then the sum (7.3.4) becomes
\[
\sum_{k=0}^{P-1} z^k = (z^P - 1)/(z - 1).
\]
Since \(l + l' \in \mathbb{Z}\) it follows that \(z^P = 1\) and the sum vanishes.

---

51
Part III

Results
8 Pure Systems

Pure molecular hydrogen (H\textsubscript{2}) and pure molecular deuterium (D\textsubscript{2}) have been simulated for two different states in the \textit{p–T} diagram, shown in figure 8.1. Also included in this figure are the saturation vapor pressure curves for H\textsubscript{2} and D\textsubscript{2} (after Roder et al. [73]). For temperatures above the critical temperature (33.18 K for H\textsubscript{2} and 38.34 K for D\textsubscript{2}) there exists only one fluid phase (a super-critical fluid). For systems below the critical temperature with \textit{p, T} values on the saturation vapor pressure curve liquid and vapor phase coexist. Above the curve only the liquid phase is stable and below the curve only the vapor phase.\textsuperscript{1}

The two state points used for the simulations were 20.4 K, 1.5 atm for state point A and 70.4 K, 747.3 atm for state point B. As it can be seen from figure 8.1 H\textsubscript{2} and D\textsubscript{2} are liquids at state point A and super-critical fluids at state point B.

The simulations were performed in the isothermal-isobaric ensemble with the Path-Integral Monte Carlo method after section 7.2. In general, the simulations consisted of \textit{N} = 108

\textsuperscript{1}For lower temperatures and/or higher pressures various solid phases come into existence.
8 Pure Systems

“quantum particles” and the Trotter number $P$ was varied between 1 and 64 (some systems were also simulated with $P = 128$). In this way it was possible to study the effects of quantum mechanics being “turned on gradually”. The intermolecular potential was assumed to be spherically symmetric of Lennard-Jones (LJ 6–12) type (see equation 6.1.1). The Lennard-Jones parameters were $\varepsilon/k_B = 36.7$ K, $\sigma = 0.296$ nm for hydrogen and $\varepsilon/k_B = 35.2$ K, $\sigma = 0.295$ nm for deuterium. The masses of H$_2$ and D$_2$ were 2.016 u ($= 3.34765 \times 10^{-27}$ kg) and 4.028 u ($= 6.68865 \times 10^{-27}$ kg), respectively. Additionally, exchange effects in the quantum-mechanical calculations were considered to be negligible and the particles were assumed to obey Boltzmann statistics. As Ceperley [17] has pointed out, even for helium at 2 K the differences in, e.g., the pair-correlation functions for systems obeying Boltzmann and for systems obeying Bose statistics is only about 2%. For higher temperatures (as used here to simulate hydrogen and deuterium) the differences should be even lower.

8.1 Lennard-Jones Systems Compared to Real Hydrogen and Deuterium

The assumption of a spherically symmetric potential for the intermolecular interactions is justified by the fact that in the ground state H$_2$ and D$_2$ have an almost spherical molecular charge distribution. According to Silvera [79] the numerical eccentricity (ratio of the minor to the major axis) of the charge density is only about 0.94 (this refers to para-H$_2$ and ortho-D$_2$). As Silvera and others [2, 44] have pointed out, the Lennard-Jones potential displays the correct $r^{-6}$ asymptotic decay of the dispersion forces at large separations but the repulsive part of the potential is incorrect at shorter distances. Another simple potential is the exp-6 potential, but the repulsive part is also incorrect [79]. Also its application to dense systems neglects many-body forces completely [44]. Nevertheless, the Lennard-Jones potential was used since it is easy to implement (especially for volume moves, see chapter 6) and captures much of the essential physics of simple liquids in a variety of applications [44]. Furthermore it is the perhaps best characterized of all potentials, real or imagined [2].

The complete phase diagrams of hydrogen and deuterium considered as classical Lennard-Jones systems are shown in figure 8.2. According to Johnson et al. [44] the critical point of a classical Lennard-Jones fluid is $T_{\text{LJ}}^c = 1.313$, $\rho_{\text{LJ}}^c = 0.13$ and $\rho_{\text{LJ}}^{\text{solid}} = 0.310$. This gives 48.2 K, 25.1 atm and 19.8 mol/l for H$_2$ and 46.2 K, 24.3 atm and 20.1 mol/l for D$_2$, respectively. Agrawal and Kofke [2] obtain for the triple point of classical Lennard-Jones systems $T_{\text{LJ}}^t = 0.687(4)$, $\rho_{\text{LJ}}^t = 0.001$ and $\rho_{\text{LJ, solid}}^t = 0.963$, $\rho_{\text{LJ, liquid}}^t = 0.850$, $\rho_{\text{LJ, vapor}}^t = 0.00186$. Using the Lennard-Jones parameters for hydrogen and deuterium one gets $T^t = 25.2$ K, $p^t = 0.193$ atm and $\rho_{\text{solid}}^t = 61.7 \text{ mol/l}$, $\rho_{\text{liquid}}^t = 54.4 \text{ mol/l}$, $\rho_{\text{vapor}}^t = 0.119 \text{ mol/l}$ for H$_2$ and $T^t = 24.2$ K, $p^t = 0.187$ atm and $\rho_{\text{solid}}^t = 62.3 \text{ mol/l}$, $\rho_{\text{liquid}}^t = 55.0 \text{ mol/l}$, $\rho_{\text{vapor}}^t = 0.120 \text{ mol/l}$ for D$_2$.

These phase diagrams have to be compared with the measured coexistence properties of

---

2The atomic mass constant has the value $m_u = 1.66053873(13) \times 10^{-27}$ kg.
3Apart from guesses from experiment, intermolecular interactions are usually determined by $ab\ initio$ calculations. A short introduction to these methods can be found in the review article by Silvera [79].
4Fitting the Lennard-Jones potential to measurement data gives an effective potential where the many-body forces are considered in an average way.
real hydrogen and deuterium. Therefore the saturation vapor pressure curves for H₂ and D₂ from figure 8.1 have been included in figure 8.2. As one can see, the diagrams for the classical Lennard-Jones systems are shifted to higher temperatures and higher pressures. Also the roles of H₂ and D₂ seem to have changed. These differences have no serious effects for the simulations at state point B (see also figure 8.3). At state point A the effects are more serious: In this state the classical Lennard-Jones systems are already solids while the real systems are still liquids. Hence for classical simulations (low Trotter numbers) one should expect solid systems and liquid systems when quantum mechanics is “turned on” (i.e., the Trotter number is increased). As figure 8.3 shows this is indeed the case.

Figure 8.3 shows the vapor-liquid coexistence curves for the classical Lennard-Jones and the real systems and the solid-liquid coexistence lines for the classical Lennard-Jones systems in the $\rho, T$ plane. The simulation results for the densities at state point A (20.4 K) and state point B (70.4 K) after tables 8.1 and 8.2 are also included. Comparing the densities for hydrogen and deuterium one can see that the density changes from $P = 1$ to $P = 64$ are larger for hydrogen. This is due to the fact that H₂ has only about half the mass of D₂. The density changes are also much larger at state point A than at state point B. Since the temperature in the latter is more than thrice the temperature at state point A it can be considered “more classical”.

Figure 8.2: Phase diagram for classical Lennard-Jones H₂ and D₂ in the $p, T$ plane (after Agrawal and Kofke [2] and Kofke [52]). The triple points (H₂: ▼, D₂: ▲) for the classical Lennard-Jones systems and the critical points (H₂: ▽, D₂: △) for the classical Lennard-Jones and the real systems are included. For comparison the saturation vapor pressure curves for real H₂ and D₂ are also shown (after Roder et al. [73])
I.e., replacing the density operator with a product of free particle density operators is a better approximation.

### 8.2 Hydrogen and Deuterium at 20.4 K and 1.5 atm

All simulations have been started from a random initial configuration. Though in isothermal-isobaric simulations the volume fluctuates one has to provide a volume to start from. These volumes have been found by short (a few hundred passes) trial-and-error simulations. For simulations with higher Trotter numbers a first guess was a slightly increased volume of a run with the preceding Trotter number.

The first phase of each simulation was equilibration of this initial configuration at constant volume. The length of this equilibration phase was some few thousand passes till the potential energy was approximately “constant”. The maximum particle displacement was adjusted to an acceptance ratio of about 50% for the classical simulations \((P = 1)\). This maximum displacement was 0.15 \(\sigma\) for the simulations in Lennard-Jones units and 0.04 nm for the simulations in physico-chemical units.

After this first equilibration the volume was allowed to change. The maximum relative
volume change was again adjusted such that the acceptance ratio for the volume moves was about 50% in the classical simulations. This maximum relative volume change was 0.06 both for the simulations in Lennard-Jones units and the simulations in physico-chemical units. Since a volume move “costs” of the order of $N$ of the simulation time of a single particle move (except for classical Lennard-Jones systems, see section 6.3.2) one makes volume moves not to frequently. A rule of thumb says that the probability for trying a volume move should be about $1/N$. After the potential energy was again approximately constant the “warm-up” phase ended. The subsequent production phase consisted of some ten blocks of $10 \times 10000$ passes each.

8.2.1 Constant Particle Number and Varying Trotter Number

Many simulations of fluid systems have been performed with $N = 108$ quantum particles. For the first simulations this particle number was held constant to observe the effects of varying the Trotter number (turning on quantum mechanics).

Simulations in Lennard-Jones Units

The pure systems have been simulated first with a program in reduced Lennard-Jones units. For some Trotter numbers these simulations have been repeated in with a program in physico-chemical units. According to the mentioned rule of thumb the probability for trying a volume move was set to 0.01. The results of these simulations for pure H$_2$ and D$_2$ at $p = 1.5$ atm and $T = 20.4$ K can be found in tables 8.1 and 8.2, respectively. For a better comparison with the mixtures the results are already converted to physico-chemical units after table 6.1. The change of the interesting properties with the Trotter number is shown in figure 8.4.

“Blocks” is the number of $10 \times 10000$ passes each simulation lasted. “Accrp” is the percentage of accepted particle moves and “accrv” is the percentage of accepted volume moves. $\sqrt{\langle r^2 \rangle}$ is the root-mean-square of the radius of gyration. $\langle \rho \rangle$, $\langle K \rangle$, $\langle U \rangle$, $\langle H \rangle = \langle K + U + pV \rangle$ and $\langle V \rangle$ are the mean-values of the density, the kinetic energy, the potential energy, the enthalpy and the volume, respectively.

For the classical simulations the acceptance ratios for particle and for volume moves are around the desired 50%. For Trotter numbers $P > 1$ accrv does not change much. The behavior of accrp is more interesting: After a sudden drop between $P = 1$ and $P = 2$ it increases slightly with the Trotter number, though the maximum displacement was kept constant for all Trotter numbers. This may have something to do with the freezing of the more classical systems. At state point B the effect is considerable smaller (but still visible).

Averaging the relative distances of the beads of a ring-polymer to the center of mass one gets the root-mean-square of the radius of gyration. It is therefore a measure of the uncertainty in the position of the molecule due to quantum mechanics. The radius of gyration increases strongly between $P = 1$ and $P = 4$ and increases only slightly from $P = 4$ to $P = 64$. This behavior of the radius of gyration is a combined effect: For larger $P$ there are more beads to distribute that cannot come closer together than approximately their Lennard-Jones diameter. On the other hand the “springs” between the beads become more stiff for higher $P$ so that the beads cannot move to far away from each other. Figure 8.4 shows how quickly the radius...
Table 8.1: Results for pure H\textsubscript{2} at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ particles. Each block of the production phase consisted of $10 \times 10\,000$ passes (e.g., $15/10$ means that only the last 15 of the 40 blocks in total have been evaluated)

<table>
<thead>
<tr>
<th>$P$</th>
<th>blocks</th>
<th>acr p</th>
<th>acc r</th>
<th>$\sqrt{\langle \tau^2 \rangle}$</th>
<th>$\langle \rho \rangle$</th>
<th>$\langle K \rangle$</th>
<th>$\langle U \rangle$</th>
<th>$\langle H \rangle$</th>
<th>$\langle V \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[%]</td>
<td>[%]</td>
<td>[pm]</td>
<td>[mol/l]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[ml/mol]</td>
</tr>
<tr>
<td>1</td>
<td>15/40</td>
<td>42.79(1)</td>
<td>48.12(4)</td>
<td>0</td>
<td>61.86(1)</td>
<td>254.4540(9)</td>
<td>-2.2468(3)</td>
<td>-1.9899(3)</td>
<td>16.164(3)</td>
</tr>
<tr>
<td>2</td>
<td>33/40</td>
<td>16.42(1)</td>
<td>53.64(3)</td>
<td>32.75(1)</td>
<td>51.59(1)</td>
<td>385.59(9)</td>
<td>-1.7510(6)</td>
<td>-1.3623(6)</td>
<td>19.386(6)</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>16.39(1)</td>
<td>57.81(2)</td>
<td>43.441(6)</td>
<td>44.54(1)</td>
<td>486.4(2)</td>
<td>-1.4443(6)</td>
<td>-0.9544(6)</td>
<td>22.462(7)</td>
</tr>
<tr>
<td>8</td>
<td>40</td>
<td>20.88(2)</td>
<td>60.25(3)</td>
<td>47.401(6)</td>
<td>40.02(2)</td>
<td>541.7(2)</td>
<td>-1.2524(9)</td>
<td>-0.7069(6)</td>
<td>25.00(1)</td>
</tr>
<tr>
<td>16</td>
<td>40</td>
<td>24.33(2)</td>
<td>60.08(2)</td>
<td>48.656(6)</td>
<td>37.87(2)</td>
<td>560.0(3)</td>
<td>-1.1624(6)</td>
<td>-0.5895(3)</td>
<td>26.43(1)</td>
</tr>
<tr>
<td>32</td>
<td>30</td>
<td>25.95(2)</td>
<td>57.35(3)</td>
<td>49.024(6)</td>
<td>37.05(2)</td>
<td>581.2(3)</td>
<td>-1.1286(6)</td>
<td>-0.5341(6)</td>
<td>27.02(1)</td>
</tr>
<tr>
<td>64</td>
<td>30</td>
<td>26.55(2)</td>
<td>52.08(3)</td>
<td>49.130(3)</td>
<td>36.76(1)</td>
<td>585.5(3)</td>
<td>-1.1170(6)</td>
<td>-0.5272(6)</td>
<td>27.22(1)</td>
</tr>
<tr>
<td>$\infty$</td>
<td></td>
<td></td>
<td></td>
<td>49.198(6)</td>
<td>36.49(3)</td>
<td>591.1(1)</td>
<td>-1.106(2)</td>
<td>-0.510(3)</td>
<td>27.44(4)</td>
</tr>
</tbody>
</table>

Table 8.2: Results for pure D\textsubscript{2} at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ particles. Each block of the production phase consisted of $10 \times 10\,000$ passes

<table>
<thead>
<tr>
<th>$P$</th>
<th>blocks</th>
<th>acr p</th>
<th>acc r</th>
<th>$\sqrt{\langle \tau^2 \rangle}$</th>
<th>$\langle \rho \rangle$</th>
<th>$\langle K \rangle$</th>
<th>$\langle U \rangle$</th>
<th>$\langle H \rangle$</th>
<th>$\langle V \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[%]</td>
<td>[%]</td>
<td>[pm]</td>
<td>[mol/l]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[ml/mol]</td>
</tr>
<tr>
<td>1</td>
<td>40</td>
<td>44.56(1)</td>
<td>49.60(2)</td>
<td>0</td>
<td>61.38(6)</td>
<td>254.383(6)</td>
<td>-2.0823(3)</td>
<td>-1.8253(3)</td>
<td>16.312(3)</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
<td>26.05(1)</td>
<td>53.78(3)</td>
<td>25.196(6)</td>
<td>53.26(1)</td>
<td>362.95(6)</td>
<td>-1.7252(6)</td>
<td>-1.3596(6)</td>
<td>18.790(6)</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>26.70(1)</td>
<td>56.25(2)</td>
<td>31.940(3)</td>
<td>48.75(1)</td>
<td>445.55(9)</td>
<td>-1.5346(6)</td>
<td>-1.0859(6)</td>
<td>20.527(6)</td>
</tr>
<tr>
<td>8</td>
<td>30</td>
<td>30.26(1)</td>
<td>57.25(3)</td>
<td>34.227(2)</td>
<td>46.29(1)</td>
<td>492.5(1)</td>
<td>-1.4307(6)</td>
<td>-0.9349(3)</td>
<td>21.62(6)</td>
</tr>
<tr>
<td>16</td>
<td>30</td>
<td>32.47(1)</td>
<td>56.46(3)</td>
<td>34.897(2)</td>
<td>45.30(1)</td>
<td>514.9(2)</td>
<td>-1.3892(6)</td>
<td>-0.8711(3)</td>
<td>22.09(5)</td>
</tr>
<tr>
<td>32</td>
<td>30</td>
<td>33.35(2)</td>
<td>53.87(3)</td>
<td>35.078(2)</td>
<td>44.99(2)</td>
<td>522.8(3)</td>
<td>-1.3760(6)</td>
<td>-0.8497(6)</td>
<td>22.25(1)</td>
</tr>
<tr>
<td>64</td>
<td>30</td>
<td>33.63(2)</td>
<td>49.30(2)</td>
<td>35.128(2)</td>
<td>44.87(1)</td>
<td>525.4(3)</td>
<td>-1.3713(6)</td>
<td>-0.8424(6)</td>
<td>22.30(5)</td>
</tr>
<tr>
<td>$\infty$</td>
<td></td>
<td></td>
<td></td>
<td>35.153(2)</td>
<td>44.7830(9)</td>
<td>527.8(3)</td>
<td>-1.36788</td>
<td>-0.8362(5)</td>
<td>22.349(2)</td>
</tr>
</tbody>
</table>
Figure 8.4: Results for pure H\textsubscript{2} and D\textsubscript{2} at $p = 1.5$ atm and $T = 20.4$ K vs. Trotter number $P$. The simulations consisted of $N = 108$ particles
of gyration converges with the Trotter number. All other properties converge more slowly in comparison. Comparing hydrogen and deuterium one can see that the radius of gyration for deuterium is only about 70% of that for hydrogen. Due to the higher mass deuterium is “more classical” than hydrogen (the “spring constant” is proportional to the particle mass).

Between $P = 1$ and $P = 8$ the densities decrease rapidly with the Trotter number. For Trotter numbers greater than 8 the densities stay approximately constant. This shows how the systems change from the solid phase for low Trotter numbers to the liquid phase for high Trotter numbers. From $P = 16$ on the systems are definitely in the liquid state. Nevertheless hydrogen seems not to have converged fully at a Trotter number of 64 while deuterium more or less has. Again there is a significant difference between H$_2$ and D$_2$: Since H$_2$ has the lower mass it is more quantum mechanical than D$_2$ and the uncertainty in its position is greater. I.e., though hydrogen is classically smaller than deuterium, it takes more space than deuterium in quantum simulations. For $P = 64$ the expectation values for the densities of hydrogen and deuterium are nearly equal to $1/\langle V \rangle$ (the factor $N$ is already included for physico-chemical units): $1/\langle V \rangle = 36.74$ mol/l for hydrogen and $1/\langle V \rangle = 44.83$ mol/l for deuterium.

The largest change of the kinetic energy is also observed between $P = 1$ and $P = 16$. After that the kinetic energy changes only slightly. Interestingly, the kinetic energy of hydrogen is larger than for deuterium by a factor of 1.11 though the latter has about twice the mass of the former. According to the equipartition theorem the kinetic energies are equal in the classical case. The differences are introduced by quantum mechanics as deuterium is more classical due to its larger mass.

The potential energy and the enthalpy behave like the density and the volume behaves like the kinetic energy. The absolute volumes of the simulation boxes increased from 2.90 nm$^3$ ($P = 1$) to 4.88 nm$^3$ ($P = 64$) for hydrogen and from 2.93 nm$^3$ to 4.00 nm$^3$ for deuterium.

According to [23, 31, 55] the partition function is accurate to order $(\beta/P)^2$. The simulation results have been fitted with $f(P) = a/P^2 + b/P + c$ and with $g(P) = a/P^2 + c$. The $y$-intercepts $c$ should give an estimation of the results for “full quantum mechanical” systems. An example for this “extrapolation” is shown in figure 8.5 for the density of deuterium at $p = 1.5$ atm and $T = 20.4$ K. The Trotter numbers have been assumed to be large enough for $P \geq 8$. As figure 8.5 shows the fit $f(P)$ with a linear term (full line) is quite good in contrast to the fit $g(P)$ with only the quadratic term (dashed line). For $P < 8$ the deviations between the simulation results and the parabolic fits through $P \geq 8$ became very large. Obviously, the dependence on the inverse Trotter number is not purely quadratic if at all. The $y$-intercept for $f(P)$ gives $\rho = 44.7830(9)$ mol/l and for $g(P)$ one gets $\rho = 44.89(4)$ mol/l. The values in the $P = \infty$ rows of tables 8.1 and 8.2 have been obtained analogously via fits of the form $f(P)$, though the fits were sometimes a little bit poorer. It should be noted that the differences between $P = 32$ and $P = 64$ are approximately the same size as the differences between $P = 64$ and $P = \infty$, contrary to what one would expect from looking at figure 8.4. It is a little bit difficult to interpret these results: The dependence on $1/P$ can be more complex and/or one is not yet in the asymptotic region.

As has been mentioned in the previous section the classical Lennard-Jones systems are in the solid phase at state point A. Though the initial configuration was a random distribution of particles in the simulation box one should expect the classical systems to freeze in after some “time”. It was really possible to observe this phase transition for small Trotter numbers in
8.2 Hydrogen and Deuterium at 20.4 K and 1.5 atm

![Graph showing polynomial fits of order 2](image)

Figure 8.5: Polynomial fits of order 2 \( f(P) = a/P^2 + b/P + c \) and \( g(P) = a/P^2 + c \) of the densities of \( \text{D}_2 \) at \( p = 1.5 \text{ atm} \) and \( T = 20.4 \text{ K} \) for \( P = 8, 16, 32 \) and 64 vs. inverse Trotter number. All simulations consisted of \( N = 108 \) quantum particles

The pair-correlation functions for \( \text{H}_2 \) (figure 8.6) and \( \text{D}_2 \) (figure 8.7). For \( P = 1 \) the resulting radial distribution function is typical for a fcc solid. This was facilitated by the fact that the number of quantum particles \( N = 108 \) is a so called “magic number” for an fcc lattice. I.e., by turning on quantum mechanics it was possible to observe a phase transition from solid to liquid at constant temperature and pressure.

Figure 8.6 shows how the pair-correlation functions for \( \text{H}_2 \) converge with the Trotter number \( P \). Though all the simulations started from random initial configurations the classical simulation shows the typical form of a pair-correlation function in the solid phase: high and narrow peaks and low minima. These characteristics are due to the high geometrical ordering in the coordination shells [19] (nearest neighbors, second nearest neighbors, …). For \( P = 2 \) and \( P = 4 \) the radial distribution functions seem to be intermediates between solids and liquids. Since the freezing took a long time it might be possible that even the systems with these two Trotter numbers would solidify in simulation runs. For \( P \geq 16 \) the pair-correlation functions are almost identical and typical for liquid systems: Because the ordering in the coordination shells is smaller for a liquid than for a solid the pair-correlation is more diffuse with smaller deviations from the bulk density. The large differences between \( P = 1 \) and \( P \geq 16 \) are mainly due to the fact that the classical systems are in the solid state and the quantum systems (the “real” systems) are in the liquid state. The quantum mechanical uncertainty in the particle positions gives an additional but much smaller reduction of the features of \( g(r) \). This can be seen figures 8.12 and 8.13 where the classical and the quantum systems are (super-critical)


Figure 8.6: Pair-correlation functions for H$_2$ for various Trotter numbers at $p = 1.5$ atm and $T = 20.4$ K. All simulations consisted of $N = 108$ quantum particles.

Table 8.3 shows the maxima and minima of the pair-correlation functions $g(r)$ of H$_2$ at state point A for various Trotter numbers. The onset distance $r_0$ is taken as the distance where $g(r)$ becomes numerically greater than zero for the first time. The definitions for the maxima ($\bar{r}_1$ and $\bar{r}_2$) and the only minimum ($\bar{r}_1$) that could be observed in the simulation boxes with $N = 108$ quantum particles posed no problems. The onset distance $r_0$ decreases from 0.253(3) nm for $P = 2$ to 0.209(3) nm for $P = 64$. Since in the quantum mechanical case the particles are more diffuse it is possible for two of them to come closer than classically allowed. This can also be seen from figure 8.6. The position of the first maximum $\tau_1$ (used as a measure for the distance of the nearest neighbors) increases from 0.324(3) nm to 0.360(3) nm. The position of the first minimum $\tau_1$ is due to the exclusion effects of the nearest and second nearest neighbor molecules and changes from 0.469(3) nm to 0.517(3) nm. The distance of the second nearest neighbors is approximated by the second maximum $\tau_2$ and increases from 0.620(3) nm to 0.682(3) nm. This shows again that the molar volume of the systems increases when quantum mechanics is turned on. Within the given accuracy the maxima and minima seem to have converged from $P = 16$ on which is comparable to the appearance in figure 8.6. Only the onsets do still decrease till $P = 64$ (and maybe even further). The results for $P = 1$ have been omitted since in the solid state additional maxima and minima emerge that do not fit into the scheme of table 8.3. For the sake of completeness the values are $r_0 = 0.265(3)$ nm, $\tau_1 = 0.324(3)$ nm, $\tau_1 = 0.416(3)$ nm, $\tau_2 = 0.475(3)$ nm, $\tau_2 = 0.505(3)$ nm, $\tau_3 = 0.576(3)$ nm and
Table 8.3: The onsets and the first maxima and minima of the pair-correlation functions of
H$_2$ for various Trotter numbers at $p = 1.5$ atm and $T = 20.4$ K. All simulations
consisted of $N = 108$ quantum particles. The statistical uncertainty is always
0.003 nm

<table>
<thead>
<tr>
<th>$P$</th>
<th>$r_0$</th>
<th>$r_1$</th>
<th>$r'_1$</th>
<th>$r_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[nm]</td>
<td>[nm]</td>
<td>[nm]</td>
<td>[nm]</td>
</tr>
<tr>
<td>2</td>
<td>0.253</td>
<td>0.324</td>
<td>0.469</td>
<td>0.620</td>
</tr>
<tr>
<td>4</td>
<td>0.241</td>
<td>0.336</td>
<td>0.493</td>
<td>0.647</td>
</tr>
<tr>
<td>8</td>
<td>0.232</td>
<td>0.351</td>
<td>0.508</td>
<td>0.670</td>
</tr>
<tr>
<td>16</td>
<td>0.221</td>
<td>0.357</td>
<td>0.514</td>
<td>0.676</td>
</tr>
<tr>
<td>32</td>
<td>0.215</td>
<td>0.360</td>
<td>0.517</td>
<td>0.682</td>
</tr>
<tr>
<td>64</td>
<td>0.209</td>
<td>0.360</td>
<td>0.517</td>
<td>0.682</td>
</tr>
</tbody>
</table>

$r_4 = 0.629(3)$ nm. It is interesting to note that the distance of the nearest neighbor molecules is
around the minimum of the Lennard-Jones potential at $\sqrt[6]{2} \sigma = 0.332$ nm for the lower Trotter
numbers. For higher $P$ the nearest neighbors are farther outwards. The same is true for the
second nearest neighbors.

The radial distribution functions for D$_2$ for $P = 1$ to $P = 64$ are shown in figure 8.7. Again,
the classical simulation results in a solid system, though not as impressive as for hydrogen.
Since the intermolecular potentials of hydrogen and deuterium are so similar this is likely to
change for longer simulation runs. The systems with $P = 2$ and $P = 4$ are again in some
intermediate state. For Trotter numbers $P \geq 8$ the pair-correlation functions seem to have
converged to a liquid system. Compared with the radial distribution functions of hydrogen
one sees that the distance of the first neighbors is almost equal. But the distance of the second
nearest neighbors is smaller for deuterium. From this one can see again that the density of
deuterium is greater than for hydrogen.

As table 8.4 shows the positions of the maxima and minima for $P = 2$ are very similar for
hydrogen and deuterium. For the larger Trotter numbers the distances are throughout smaller
then for hydrogen. In the case of hydrogen the distances of the nearest and second nearest
neighbors are larger than the expected values of $\sqrt[6]{2} \sigma = 0.332$ nm and $2\sqrt[6]{2} \sigma = 0.664$ nm. For
deuterium the distance of the nearest neighbors is larger than the expected value of $\sqrt[6]{2} \sigma =
0.331$ nm. However, the second nearest neighbors are nearer than expected ($2\sqrt[6]{2} \sigma = 0.662$ nm).

**Simulations in Physico-Chemical Units**

For simulations of mixtures it is easier to use physico-chemical than Lennard-Jones units in the
simulation programs. For $P = 1, 4, 64$ and 128 the simulations of the previous section have been
repeated with a program using physico-chemical units. Additionally, it was studied to what
extent the simulation results depend on the different masses and the different intermolecular
potentials of hydrogen and deuterium. To do this the hypothetical molecules H$_2^*$ and D$_2^*$ were
introduced. H$_2^*$ is the mass of hydrogen with the Lennard-Jones parameters of deuterium and
Figure 8.7: Pair-correlation functions for $D_2$ for various Trotter numbers at $p = 1.5\, \text{atm}$ and $T = 20.4\, \text{K}$. All simulations consisted of $N = 108$ quantum particles

$D_2^*$ is analogous the other way around. The production phases of all simulations consisted of 20 blocks of $10 \times 10000$ passes each.

The simulation results are listed in table 8.5. The differences in the results between tables 8.1 and 8.2 and table 8.5 are due to the fact that the number of blocks in the simulations were different. Additionally, the input parameters of the two programs were not exactly equal due to rounding.

Since the mass enters the simulations only for Trotter numbers $P > 1$ the classical simulations should give nearly identical results for $H_2$ and $D_2^*$ and $D_2$ and $H_2^*$. As table 8.5 shows this is approximately fulfilled. The small differences are due to different warm up lengths. For higher Trotter numbers the influence of the different masses overwhelms the differences in the intermolecular potentials, so $H_2$ and $H_2^*$ and $D_2$ and $D_2^*$ become more similar. Besides these differences the variation of the simulation results with the Trotter number is the same as for the simulations in Lennard-Jones units in the previous section.

Figure 8.8 compares the pair-correlation functions of $H_2$, $H_2^*$, $D_2$ and $D_2^*$ for a Trotter number of $P = 64$. Though there are slight differences between $H_2$ and $H_2^*$ and $D_2$ and $D_2^*$ the different masses have obviously the greatest influence on the radial distribution functions. Additionally, systems with the Lennard-Jones parameters of $D_2$ seem to have a little less pronounced structure than systems with the parameters for $H_2$. This is due to the fact that the Lennard-Jones potential for hydrogen has a deeper energy minimum than for deuterium ($36.7\, \text{K}$ compared to $35.2\, \text{K}$). This interpretation is also supported by the fact that the differences in the pair-correlation functions become smaller for larger distances. For $P = 1$ the pair-correlation
functions for $D_2$ and $H^*_2$ and for $H_2$ and $D_2^*$ are very similar since in the classical simulations the different masses have no effect. Again the structure of the hydrogen-like systems is more pronounced than for the deuterium-like ones. Comparing the radial distribution functions between the simulations in Lennard-Jones and in physico-chemical units the differences are smaller than the line widths.

### 8.2.2 Constant Trotter Number and Varying Particle Number

Since all computer simulations consist of a very small number of particles compared to the macroscopic systems finite size effects affect the simulation results. To get an estimation of these finite size effects the Trotter number was held constant at $P = 64$ and the number of quantum particles $N$ was varied between $N = 16$ and $N = 256$. The probability for a volume change was held constant at 0.01 independent of $N$.

The simulations have been performed in Lennard-Jones units. The results have been converted to physico-chemical units and are shown in tables 8.6 and 8.7 for hydrogen and deuterium, respectively. The results vs. the number of quantum particles are graphically shown in figure 8.9.

Though for the simulation of $H_2$ with $N = 16$ the system “evaporated” ($\rho \rightarrow 0$), the number of quantum particles seems to have only a very small effect on the simulation results. For $D_2$ the systems should consist at least of $N = 64$ particles. Though a larger $N$ is always preferable the simulations with $N = 108$ seem to be a good compromise between accuracy and time and memory considerations. Since the run-times are at least proportional to $N^2$ (for fixed $P$) the simulations with $N = 256$ particles were stopped after only a few blocks. Nevertheless, the statistics should be as good as for a system with $N = 108$ particles and the doubled run-times.

The pair-correlation functions have always been recorded up to half the box length of the cubical simulation box. Since the box length increases with the number of particles the radial distribution functions for $N = 256$ include also the third nearest neighbors. The results for $P = 64$ are shown in figure 8.10. For $D_2$ the first maxima are at $r_1 = 0.344(3)$ nm, $r_2 =$
Table 8.5: Results for pure H$_2$, H$_2^*$, D$_2$ and D$_2^*$ at $p = 1.5$ atm and $T = 20.4$ K with $N = 108$ particles. Each simulation consisted of a production phase of 20 blocks of $10 \times 10,000$ passes each.

<table>
<thead>
<tr>
<th>$P$</th>
<th>accrp [%]</th>
<th>accrv [%]</th>
<th>$\sqrt{\langle r^2 \rangle}$ [pm]</th>
<th>$\langle \rho \rangle$ [mol/l]</th>
<th>$\langle K \rangle$ [kJ/mol]</th>
<th>$\langle U \rangle$ [kJ/mol]</th>
<th>$\langle H \rangle$ [kJ/mol]</th>
<th>$\langle V \rangle$ [ml/mol]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>47.61(4)</td>
<td>48.29(5)</td>
<td>0</td>
<td>61.31(7)</td>
<td>254.404(2)</td>
<td>-2.200(4)</td>
<td>-1.943(4)</td>
<td>16.31(2)</td>
</tr>
<tr>
<td>4</td>
<td>16.92(2)</td>
<td>57.46(5)</td>
<td>43.42(1)</td>
<td>44.59(2)</td>
<td>486.8(3)</td>
<td>-1.4458(9)</td>
<td>-0.9556(7)</td>
<td>22.44(1)</td>
</tr>
<tr>
<td>64</td>
<td>27.07(2)</td>
<td>51.87(4)</td>
<td>49.091(5)</td>
<td>36.93(2)</td>
<td>588.0(6)</td>
<td>-1.1220(7)</td>
<td>-0.5298(6)</td>
<td>27.11(1)</td>
</tr>
<tr>
<td>128</td>
<td>27.29(2)</td>
<td>44.19(4)</td>
<td>49.128(6)</td>
<td>36.81(2)</td>
<td>588.4(6)</td>
<td>-1.1172(7)</td>
<td>-0.5247(6)</td>
<td>27.19(1)</td>
</tr>
<tr>
<td>1</td>
<td>48.40(4)</td>
<td>49.06(6)</td>
<td>0</td>
<td>61.94(6)</td>
<td>254.404(2)</td>
<td>-2.119(5)</td>
<td>-1.862(5)</td>
<td>16.15(2)</td>
</tr>
<tr>
<td>4</td>
<td>18.70(2)</td>
<td>58.82(3)</td>
<td>43.982(9)</td>
<td>43.90(3)</td>
<td>473.8(2)</td>
<td>-1.342(1)</td>
<td>-0.8649(8)</td>
<td>22.80(1)</td>
</tr>
<tr>
<td>64</td>
<td>29.64(3)</td>
<td>52.71(4)</td>
<td>49.470(5)</td>
<td>35.98(3)</td>
<td>562.5(5)</td>
<td>-1.0292(9)</td>
<td>-0.4625(6)</td>
<td>27.83(2)</td>
</tr>
<tr>
<td>1</td>
<td>48.403(4)</td>
<td>49.04(4)</td>
<td>0</td>
<td>61.958(3)</td>
<td>254.404(2)</td>
<td>-2.1147(1)</td>
<td>-1.8578(1)</td>
<td>16.1429(9)</td>
</tr>
<tr>
<td>4</td>
<td>27.91(2)</td>
<td>55.90(4)</td>
<td>31.920(4)</td>
<td>48.80(2)</td>
<td>446.1(1)</td>
<td>-1.5367(8)</td>
<td>-1.0875(7)</td>
<td>20.501(8)</td>
</tr>
<tr>
<td>64</td>
<td>35.08(2)</td>
<td>49.08(3)</td>
<td>35.120(2)</td>
<td>44.89(2)</td>
<td>525.4(4)</td>
<td>-1.3720(7)</td>
<td>-0.8432(7)</td>
<td>22.289(9)</td>
</tr>
<tr>
<td>128</td>
<td>35.18(2)</td>
<td>42.44(4)</td>
<td>35.133(2)</td>
<td>44.86(2)</td>
<td>526.8(4)</td>
<td>-1.3709(7)</td>
<td>-0.8407(8)</td>
<td>22.30(1)</td>
</tr>
<tr>
<td>1</td>
<td>47.704(7)</td>
<td>48.42(4)</td>
<td>0</td>
<td>61.148(8)</td>
<td>254.404(2)</td>
<td>-2.1900(4)</td>
<td>-1.9331(4)</td>
<td>16.356(2)</td>
</tr>
<tr>
<td>4</td>
<td>26.11(2)</td>
<td>54.75(5)</td>
<td>31.628(5)</td>
<td>49.28(2)</td>
<td>455.9(2)</td>
<td>-1.6436(9)</td>
<td>-1.1846(7)</td>
<td>20.302(8)</td>
</tr>
<tr>
<td>64</td>
<td>33.14(2)</td>
<td>48.36(4)</td>
<td>34.924(2)</td>
<td>45.40(2)</td>
<td>542.4(4)</td>
<td>-1.4702(7)</td>
<td>-0.9245(9)</td>
<td>22.036(9)</td>
</tr>
</tbody>
</table>
8.2 Hydrogen and Deuterium at $20.4\, K$ and $1.5\, atm$

Table 8.6: Results for pure $H_2$ at $p = 1.5\, atm$ and $T = 20.4\, K$ with a Trotter number $P = 64$. Each block of the production phase consisted of $10 \times 10^4$ passes

<table>
<thead>
<tr>
<th>$N$</th>
<th>blocks</th>
<th>acrpr</th>
<th>accrv</th>
<th>$\sqrt{\langle r^2 \rangle}$</th>
<th>$\langle \rho \rangle$</th>
<th>$\langle K \rangle$</th>
<th>$\langle U \rangle$</th>
<th>$\langle H \rangle$</th>
<th>$\langle V \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>20</td>
<td>68.64(5)</td>
<td>49.14(1)</td>
<td>36.77(5)</td>
<td>585.8(9)</td>
<td>-1.114(2)</td>
<td>-0.524(2)</td>
<td>27.33(4)</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>20</td>
<td>61.13(6)</td>
<td>49.207(9)</td>
<td>36.43(4)</td>
<td>581.2(6)</td>
<td>-1.105(2)</td>
<td>-0.519(1)</td>
<td>27.51(2)</td>
<td></td>
</tr>
<tr>
<td>108</td>
<td>30</td>
<td>52.08(3)</td>
<td>49.130(3)</td>
<td>36.76(1)</td>
<td>585.5(3)</td>
<td>-1.1170(6)</td>
<td>-0.5272(6)</td>
<td>27.22(1)</td>
<td></td>
</tr>
<tr>
<td>256</td>
<td>5</td>
<td>36.68(6)</td>
<td>49.133(6)</td>
<td>36.69(2)</td>
<td>585.5(3)</td>
<td>-1.1142(9)</td>
<td>-0.5248(9)</td>
<td>27.26(1)</td>
<td></td>
</tr>
</tbody>
</table>

Table 8.7: Results for pure $D_2$ at $p = 1.5\, atm$ and $T = 20.4\, K$ with a Trotter number $P = 64$. Each block of the production phase consisted of $10 \times 10^4$ passes (17/20 means that only the last 17 of the 20 blocks in total have been evaluated)

<table>
<thead>
<tr>
<th>$N$</th>
<th>blocks</th>
<th>acrpr</th>
<th>accrv</th>
<th>$\sqrt{\langle r^2 \rangle}$</th>
<th>$\langle \rho \rangle$</th>
<th>$\langle K \rangle$</th>
<th>$\langle U \rangle$</th>
<th>$\langle H \rangle$</th>
<th>$\langle V \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>20</td>
<td>60.88(7)</td>
<td>33.07(1)</td>
<td>54.89(5)</td>
<td>688(1)</td>
<td>-1.879(3)</td>
<td>-1.188(2)</td>
<td>18.26(2)</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>17/20</td>
<td>65.2(1)</td>
<td>34.31(6)</td>
<td>50.8(5)</td>
<td>585(6)</td>
<td>-1.62(2)</td>
<td>-1.03(2)</td>
<td>19.8(2)</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>20</td>
<td>58.63(3)</td>
<td>35.199(3)</td>
<td>44.29(3)</td>
<td>521.0(6)</td>
<td>-1.347(1)</td>
<td>-0.823(1)</td>
<td>22.61(1)</td>
<td></td>
</tr>
<tr>
<td>108</td>
<td>30</td>
<td>49.30(2)</td>
<td>35.128(2)</td>
<td>44.87(1)</td>
<td>525.4(3)</td>
<td>-1.3713(6)</td>
<td>-0.8424(6)</td>
<td>22.305(7)</td>
<td></td>
</tr>
<tr>
<td>256</td>
<td>7</td>
<td>34.12(5)</td>
<td>35.158(3)</td>
<td>44.55(2)</td>
<td>523.6(3)</td>
<td>-1.3605(9)</td>
<td>-0.8333(6)</td>
<td>22.46(1)</td>
<td></td>
</tr>
</tbody>
</table>
Figure 8.8: Pair-correlation functions for $H_2$, $H^*_2$, $D_2$ and $D^*_2$ at $p = 1.5$ atm and $T = 20.4$ K. All simulations consisted of $N = 108$ quantum particles and $P = 64$ beads per particle.

0.648(3) nm and $r_3 = 0.940(3)$ nm, the minima are at $r_1 = 0.488(3)$ nm and $r_2 = 0.792(3)$ nm. For $H_2$ are the first maxima at $r_1 = 0.360(3)$ nm, $r_2 = 0.682(3)$ nm and $r_3 = 0.996$ nm, the minima are at $r_1 = 0.517(3)$ nm and $r_2 = 0.836(3)$ nm. The distances of the nearest and second nearest neighbors are in agreement with the simulations with $N = 108$ quantum particles and $P = 64$ beads per particle. The third maxima are already very small so that one can assume that the molecules see a continuum with a constant density for distances farther than the third nearest neighbors.

The coordination number (approximately the number of nearest neighbors) is given by

$$n_{NN} = 4\pi \rho \int_0^{r_1} r^2 g(r) \, dr,$$

(8.2.1)

where $r_1$ is the first minimum of the pair-correlation function $g(r)$. After Chandler [19] it should be around 12 for liquids and solids (12 is the exact number of nearest neighbors for an fcc lattice). With the values for the first minima and the pair-correlation functions from figure 8.10 the coordination number for hydrogen yields $n_{NN} = 12.3$ and for deuterium $n_{NN} = 12.6$.

### 8.3 Hydrogen and Deuterium at 70.4 K and 747.3 atm

The simulations at state point B have been started like the ones at state point A (see section 8.2). To achieve again a acceptance ratio of about 50% for the particle and volume moves
Figure 8.9: Results for pure H$_2$ and D$_2$ at $p = 1.5$ atm and $T = 20.4$ K with a Trotter number $P = 64$ vs. particle number $N$. 

8.3 Hydrogen and Deuterium at 70.4 K and 747.3 atm
Figure 8.10: Pair-correlation functions for $H_2$ and $D_2$ at $p = 1.5 \text{ atm}$ and $T = 20.4 \text{ K}$. The simulations consisted of $N = 256$ quantum particles and $P = 64$ beads per particle.

in the classical systems the maximum displacement was set to $0.3 \sigma$ for the simulations in Lennard-Jones units and 0.09 nm for the simulations in physico-chemical units. The maximum relative volume change was set to 0.1 both for the simulations in Lennard-Jones units and in physico-chemical units. The production phases consisted again of some ten blocks of $10 \times 10^4$ passes each.

8.3.1 Constant Particle Number and Varying Trotter Number

At state point B only the dependence on the variation of the Trotter number was studied and the particle number was fixed at $N = 108$ quantum particles. According to the rule of thumb of section 8.2 the probability for a volume change was set to 0.01.

Simulations in Lennard-Jones Units

At first the simulations were again performed in Lennard-Jones units. The results for $H_2$ and $D_2$ at $p = 747.3 \text{ atm}$ and $T = 70.4 \text{ K}$ are listed in tables 8.8 and 8.9, respectively. The results are already converted to physico-chemical units after table 6.1 on page 45. Figure 8.11 shows the simulation results plotted against the Trotter number.

The values for $P = \infty$ in tables 8.8 and 8.9 have been obtained according to the method described in section 8.2.1. Again the differences between $P = 32$ and $P = 64$ are approximately the same size as the differences between $P = 64$ and $P = \infty$ in contrast to the appearance of figure 8.11. The interpretation is the same as for the systems at state point A in section 8.2.1.
Table 8.8: Results for pure H\textsubscript{2} at \( p = 747.3 \text{ atm} \) and \( T = 70.4 \text{ K} \) with \( N = 108 \) particles. Each block of the production phase consisted of \( 10 \times 10000 \) passes

\[
\begin{array}{cccccccccc}
P & \text{blocks} & \text{accep} & \text{accrc} & \sqrt{\langle r^2 \rangle} & \langle \rho \rangle & \langle K \rangle & \langle U \rangle & \langle H \rangle & \langle V \rangle \\
& & \% & \% & \text{[pm]} & \text{[mol/l]} & \text{[kJ/mol]} & \text{[kJ/mol]} & \text{[kJ/mol]} & \text{[ml/mol]} \\
1 & 30 & 44.35(8) & 50.17(3) & 0 & 48.61(6) & 0.87(7) & -1.416(3) & 1.019(6) & 20.579(3) \\
2 & 30 & 40.36(6) & 50.93(3) & 23.01(2) & 45.43(5) & 1.03(2) & -1.265(3) & 1.43(18) & 22.0(3) \\
4 & 30 & 41.16(5) & 50.79(3) & 26.80(9) & 43.74(4) & 1.13(2) & -1.18(2) & 1.68(14) & 22.8(9) \\
8 & 30 & 42.15(1) & 49.59(3) & 27.86(1) & 43.03(6) & 1.18(5) & -1.14(5) & 1.79(8) & 23.2(4) \\
16 & 30 & 42.61(8) & 46.98(3) & 28.15(9) & 42.79(6) & 1.20(2) & -1.13(6) & 1.84(13) & 23.3(7) \\
32 & 20 & 42.79(7) & 42.51(4) & 28.22(1) & 42.71(6) & 1.21(5) & -1.13(1) & 1.85(3) & 23.4(2) \\
64 & 30 & 42.86(1) & 36.26(3) & 28.24(0) & 42.67(6) & 1.21(8) & -1.13(7) & 1.8(1) & 23.4(4) \\
\infty & & 28.24(8) & 42.64(3) & 1.2195(8) & -1.13(0) & 1.86(4) & 23.45(7) & & \\
\end{array}
\]

Table 8.9: Results for pure D\textsubscript{2} at \( p = 747.3 \text{ atm} \) and \( T = 70.4 \text{ K} \) with \( N = 108 \) particles. Each block of the production phase consisted of \( 10 \times 10000 \) passes

\[
\begin{array}{cccccccccc}
P & \text{blocks} & \text{accep} & \text{accrc} & \sqrt{\langle r^2 \rangle} & \langle \rho \rangle & \langle K \rangle & \langle U \rangle & \langle H \rangle & \langle V \rangle \\
& & \% & \% & \text{[pm]} & \text{[mol/l]} & \text{[kJ/mol]} & \text{[kJ/mol]} & \text{[kJ/mol]} & \text{[ml/mol]} \\
1 & 40 & 45.36(6) & 50.58(2) & 0 & 48.71(5) & 0.87(1) & -1.33(0) & 1.10(6) & 20.55(3) \\
2 & 40 & 43.88(7) & 51.23(3) & 16.83(2) & 46.67(5) & 0.98(0) & -1.24(13) & 1.36(3) & 21.44(3) \\
4 & 40 & 44.55(6) & 50.77(2) & 19.36(2) & 45.70(5) & 1.03(8) & -1.19(0) & 1.49(6) & 21.87(3) \\
8 & 30 & 45.05(6) & 49.45(3) & 20.02(1) & 45.42(5) & 1.06(3) & -1.18(3) & 1.54(3) & 22.04(3) \\
16 & 30 & 45.23(6) & 46.72(3) & 20.19(5) & 45.32(6) & 1.07(19) & -1.17(8) & 1.56(9) & 22.08(3) \\
32 & 20 & 45.31(1) & 42.36(4) & 20.23(9) & 45.27(6) & 1.07(36) & -1.17(9) & 1.57(1) & 22.10(9) \\
64 & 30 & 45.34(8) & 36.12(3) & 20.25(9) & 45.25(6) & 1.07(4) & -1.17(8) & 1.57(2) & 22.12(3) \\
\infty & & 20.25(6) & 45.23(4) & 1.07(36) & -1.17(2) & 1.57(36) & 22.13(9) & & \\
\end{array}
\]
Figure 8.11: Results for pure H$_2$ and D$_2$ at $p = 747.3$ atm and $T = 70.4$ K with $N = 108$ particles vs. Trotter number $P$. 

8 Pure Systems
Comparing the results for state point B with state point A one can see that the differences in the results between \( P = 1 \) and \( P = 64 \) are in general dramatically smaller for state point B than for state point A.

The acceptance ratio for particle moves is nearly constant in state B. Nevertheless, a small drop between \( P = 1 \) and \( P = 2 \) and a slight increase for \( P > 2 \) can still be observed, though this effect is considerably smaller than at state point A. In contrast to state A the acceptance ratio for volume moves accrves decreases with increasing Trotter number.

The behavior of the radii of gyration is similar to state A. Since state B is a more classical state the radii of gyration at \( P = 64 \) are both smaller by a factor of about 1.75 compared to \( T = 20.4 \text{ K} \). Compared with hydrogen the radius of gyration for deuterium is again smaller by a factor of about 1.40. At both state points deuterium is more classical than hydrogen since it has about twice the mass.

That state B is more classical than state A can be seen above all in the densities of the systems. At state point A the classical systems are solids while the systems with higher Trotter numbers are liquids. This causes a large change of the densities in the transition from \( P = 1 \) to \( P = 64 \). At state point B the classical and quantum mechanical systems are supercritical fluids and the state is in general more classical (due to its higher temperature). In state B the density changes between \( P = 1 \) and \( P = 64 \) are about \(-12.2\% \) for hydrogen and about \(-7.1\% \) for deuterium. At state point A the corresponding changes are \(-40.6\% \) and \(-26.9\% \). In both cases deuterium is more classical. It is interesting to note that the densities at \( p = 1.5 \text{ atm}, T = 20.4 \text{ K} \) and \( p = 747.3 \text{ atm}, T = 70.4 \text{ K} \) are nearly the same for D\(_2\). For H\(_2\) the density increases by a factor of 1.16 between the states A and B. Obviously, hydrogen is more compressible than deuterium. Again \( 1/\langle V \rangle \) is almost equal to the expectation value for the density: \( 1/\langle V \rangle = 42.66 \text{ mol} / \text{l} \) for hydrogen and \( 1/\langle V \rangle = 45.21 \text{ mol} / \text{l} \) for deuterium.

For the classical systems at state point B the kinetic energies are by a factor \( 70.4/20.5 \approx 3.45 \) larger than in state A, as required by the equipartition theorem. For \( P = 64 \) the kinetic energies for H\(_2\) and D\(_2\) are approximately doubled. For \( P = 64 \) the potential energies for hydrogen are almost the same in the two states. For deuterium the potential energy is decreased by a factor of about 0.86. At state point B the potential energies for hydrogen and deuterium are more similar than at state point A (see also figure 8.11). For the kinetic energies it is the other way around. Between state A and B the enthalpies for hydrogen and deuterium increase by about 2.40 kJ/mol \( (P = 64) \). Hence the differences between H\(_2\) and D\(_2\) remain at about 300 J/mol.

As for the densities the volumes of hydrogen and deuterium are more similar at state point B. Again H\(_2\) seems to be not yet converged for a Trotter number of \( P = 64 \).

Figure 8.12 shows the pair-correlation functions for H\(_2\) at \( p = 747.3 \text{ atm and } T = 70.4 \text{ K} \) for various Trotter numbers. In contrast to the same functions at \( p = 1.5 \text{ atm and } T = 20.4 \text{ K} \) the differences between \( P = 1 \) and \( P = 64 \) are rather small. The radial distribution function \( g(r) \) seems already to be converged for Trotter numbers \( P \geq 8 \). The distances of the maxima and minima of the pair-correlation functions increase with the Trotter number (see table 8.10) since the particles need more space and the volume increases. But the spring constants of the springs connecting two consecutive beads also increase (the springs become more stiff) with increasing Trotter number. These two processes work against each other such that they compensate from some Trotter number on.

The onset distances and the distances of the first maxima and minima for hydrogen at
state point B are listed in table 8.10 for Trotter numbers \(1 < P < 64\). As can be also seen from figure 8.12 the distances of the maxima and minima have already converged within the statistical error for \(P \geq 8\). The distance of the nearest neighbors is about the expected \(\sqrt{2}\sigma = 0.332\) nm. The second nearest neighbors are nearer than expected \(\sqrt{2}\sigma = 0.664\) nm. Compared with the values at state point A the distances of the nearest and second nearest neighbors are smaller. This is due to the higher pressure in state B.

The pair-correlation functions for deuterium at \(p = 747.3\) atm and \(T = 70.4\) K are shown in figure 8.13. Since due to its higher mass deuterium is more classical the radial distribution functions seem to have converged even for \(P \geq 4\). This is in agreement with the general convergence of the system properties in table 8.9 and figure 8.11.

The onsets and the first maxima and minima of the radial distribution functions of \(D_2\) at state point B are given in table 8.11. Again one sees that the distances of the maxima and minima have converged within the statistical accuracy for \(P \geq 4\). Even the onsets do not change for \(P \geq 8\). Overall, one can say that \(D_2\) at \(p = 747.3\) atm and \(T = 70.4\) K is nearly a classical fluid.

Figure 8.14 compares the pair-correlation functions of hydrogen and deuterium for the two studied state points. At state point A the radial distribution functions for hydrogen and deuterium are rather different. In state B the pair-correlation functions of \(H_2\) and \(D_2\) are very similar. Since the temperature in state B is more than three times the temperature at state A the systems are more classical such that the different masses are no longer important (at least for particles with so similar potentials). Also the distances of the nearest and second nearest
Table 8.10: The onsets and the first maxima and minima of the pair-correlation functions of H\textsubscript{2} for various Trotter numbers at \( p = 747.3 \) atm and \( T = 70.4 \) K. All simulations consisted of \( N = 108 \) quantum particles. The statistical uncertainty is always 0.003 nm

<table>
<thead>
<tr>
<th>( P )</th>
<th>( r_0 )</th>
<th>( r_1 )</th>
<th>( r_1' )</th>
<th>( r_2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.241</td>
<td>0.312</td>
<td>0.463</td>
<td>0.614</td>
</tr>
<tr>
<td>2</td>
<td>0.235</td>
<td>0.321</td>
<td>0.478</td>
<td>0.626</td>
</tr>
<tr>
<td>4</td>
<td>0.223</td>
<td>0.330</td>
<td>0.484</td>
<td>0.632</td>
</tr>
<tr>
<td>8</td>
<td>0.215</td>
<td>0.333</td>
<td>0.487</td>
<td>0.641</td>
</tr>
<tr>
<td>16</td>
<td>0.209</td>
<td>0.333</td>
<td>0.487</td>
<td>0.638</td>
</tr>
<tr>
<td>32</td>
<td>0.203</td>
<td>0.333</td>
<td>0.487</td>
<td>0.638</td>
</tr>
<tr>
<td>64</td>
<td>0.200</td>
<td>0.333</td>
<td>0.487</td>
<td>0.641</td>
</tr>
</tbody>
</table>

neighbors are smaller for state B due to the higher pressure.

The coordination numbers \( n_{NN} \) after Chandler [19] (see (8.2.1)) are \( n_{NN} = 12.3 \) and \( n_{NN} = 12.7 \) for hydrogen and deuterium at 1.5 atm and 20.4 K, respectively. For the systems at 747.3 atm and 70.4 K the results are \( n_{NN} = 11.9 \) for hydrogen and \( n_{NN} = 11.8 \) for deuterium. The systems at state point A are obviously more ordered than the systems at state point B as was to be expected.

**Simulations in Physico-Chemical Units**

For Trotter numbers \( P = 1, 4 \) and 64 the simulations have been repeated with a program using physico-chemical units. The results are shown in table 8.12. As for state point A the production phases for all simulations consisted of 20 blocks of 10 \times 10000 passes each.

Again there are slight differences between the simulation results in Lennard-Jones and in physico-chemical units. These are due to different run times and different rounding errors in the simulations. Apart from these small discrepancies the simulations in Lennard-Jones and in physico-chemical units are similar.
Figure 8.13: Pair-correlation functions for $D_2$ for various Trotter numbers at $p = 747.3\text{ atm}$ and $T = 70.4\text{ K}$. All simulations consisted of $N = 108$ quantum particles.

Table 8.11: The onsets and the first maxima and minima of the pair-correlation functions of $D_2$ for various Trotter numbers at $p = 747.3\text{ atm}$ and $T = 70.4\text{ K}$. All simulations consisted of $N = 108$ quantum particles. The statistical uncertainty is always $0.003\text{ nm}$.

<table>
<thead>
<tr>
<th>$P$</th>
<th>$r_0$ [nm]</th>
<th>$\tau_1$ [nm]</th>
<th>$\xi_1$ [nm]</th>
<th>$\tau_2$ [nm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.243</td>
<td>0.314</td>
<td>0.468</td>
<td>0.612</td>
</tr>
<tr>
<td>2</td>
<td>0.235</td>
<td>0.317</td>
<td>0.473</td>
<td>0.618</td>
</tr>
<tr>
<td>4</td>
<td>0.226</td>
<td>0.323</td>
<td>0.473</td>
<td>0.624</td>
</tr>
<tr>
<td>8</td>
<td>0.217</td>
<td>0.323</td>
<td>0.476</td>
<td>0.627</td>
</tr>
<tr>
<td>16</td>
<td>0.211</td>
<td>0.323</td>
<td>0.476</td>
<td>0.627</td>
</tr>
<tr>
<td>32</td>
<td>0.211</td>
<td>0.326</td>
<td>0.476</td>
<td>0.627</td>
</tr>
<tr>
<td>64</td>
<td>0.211</td>
<td>0.326</td>
<td>0.476</td>
<td>0.627</td>
</tr>
</tbody>
</table>
8.3 Hydrogen and Deuterium at 70.4 K and 747.3 atm

Figure 8.14: Comparison of the pair-correlation functions of hydrogen and deuterium at $p = 1.5$ atm, $T = 20.4$ K and $p = 747.3$ atm, $T = 70.4$ K, respectively. The simulations consisted of $N = 108$ quantum particles and $P = 64$ beads per particle.
Table 8.12: Results for pure H$_2$ and D$_2$ at $p = 747.3$ atm and $T = 70.4$ K with $N = 108$ particles. Each simulation consisted of a production phase of 20 blocks of $10 \times 10,000$ passes each.

<table>
<thead>
<tr>
<th>$P$</th>
<th>accrp</th>
<th>accrv</th>
<th>$\sqrt{\langle v^2 \rangle}$</th>
<th>$\langle \rho \rangle$</th>
<th>$\langle K \rangle$</th>
<th>$\langle U \rangle$</th>
<th>$\langle H \rangle$</th>
<th>$\langle V \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[%]</td>
<td>[%]</td>
<td>[pm]</td>
<td>[mol/l]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[ml/mol]</td>
</tr>
<tr>
<td>H$_2$</td>
<td>1 44.032(8)</td>
<td>50.09(3)</td>
<td>0</td>
<td>48.480(8)</td>
<td>0.878000(1)</td>
<td>−1.4123(3)</td>
<td>1.0288(5)</td>
<td>20.644(3)</td>
</tr>
<tr>
<td></td>
<td>4 40.966(8)</td>
<td>50.70(4)</td>
<td>26.819(1)</td>
<td>43.621(5)</td>
<td>1.1315(3)</td>
<td>−1.1804(2)</td>
<td>1.6884(5)</td>
<td>22.944(3)</td>
</tr>
<tr>
<td></td>
<td>64 42.615(9)</td>
<td>36.21(5)</td>
<td>28.244(1)</td>
<td>42.569(9)</td>
<td>1.214(1)</td>
<td>−1.1295(3)</td>
<td>1.865(1)</td>
<td>23.511(5)</td>
</tr>
<tr>
<td>D$_2$</td>
<td>1 44.906(6)</td>
<td>50.77(4)</td>
<td>0</td>
<td>48.532(6)</td>
<td>0.878000(1)</td>
<td>−1.3306(2)</td>
<td>1.1090(4)</td>
<td>20.622(2)</td>
</tr>
<tr>
<td></td>
<td>4 44.151(4)</td>
<td>50.59(4)</td>
<td>19.365(1)</td>
<td>45.598(4)</td>
<td>1.0378(3)</td>
<td>−1.1960(2)</td>
<td>1.5037(4)</td>
<td>21.949(2)</td>
</tr>
<tr>
<td></td>
<td>64 44.925(9)</td>
<td>36.12(4)</td>
<td>20.2532(6)</td>
<td>45.108(6)</td>
<td>1.071(1)</td>
<td>−1.1740(2)</td>
<td>1.577(1)</td>
<td>22.188(3)</td>
</tr>
</tbody>
</table>
9 Mixtures

Various mixtures of hydrogen and deuterium have been simulated for the same state points as the pure systems, i.e., \( p = 1.5 \text{ atm} \), \( T = 20.4 \text{ K} \) (state A) and \( p = 747.3 \text{ atm} \), \( T = 70.4 \text{ K} \) (state B). These mixtures have been compared with ideal mixtures. The ultimately interesting results were the excess properties, specifically the excess enthalpies and the excess volumes.

The simulations consisted of \( N = 108 \) quantum particles and the Trotter number was varied between \( P = 1 \) and \( P = 64 \). Mixtures with a mole fraction of hydrogen \( \chi_{\text{H}_2} = 30\% \), \( 50\% \) and \( 70\% \) were studied. The 50\% mixture at state point A was additionally studied with a Trotter number of \( P = 128 \).

In simulations of hydrogen/deuterium mixtures one has six different Lennard-Jones parameters and two different masses. Though it would be possible to run mixture programs in Lennard-Jones units this is somewhat awkward: One of the constituents would have to be chosen as the reference and the properties of the other constituent would have to be given relative to this reference. By choosing physico-chemical units this problem is avoided. Hence the mixtures were only simulated in physico-chemical units. (To be directly comparable with the results for the pure systems the latter have also been simulated in physico-chemical for some Trotter numbers).

9.1 Intermolecular Potential for the Interaction of Hydrogen and Deuterium

The interaction between hydrogen and deuterium was also assumed to be of Lennard-Jones 6–12 type. The Lennard-Jones parameters were calculated using the Lorentz-Berthelot combining rules

\[
\begin{align*}
\varepsilon_{\text{H}_2\rightarrow\text{D}_2} &= \sqrt{\varepsilon_{\text{H}_2}\varepsilon_{\text{D}_2}}, \\
\sigma_{\text{H}_2\rightarrow\text{D}_2} &= \frac{\sigma_{\text{H}_2} + \sigma_{\text{D}_2}}{2}.
\end{align*}
\]

(9.1.1)

As it can be seen from the article of Challa and Johnson [18] for mixtures of hydrogen and neon the model potential between the components of a mixture can have a great influence on the results (especially the excess properties). Since hydrogen and deuterium have nearly the same intermolecular potential the Lorentz-Berthelot combining rules should give a reasonable potential for the hydrogen-deuterium interaction. Since the parameters of the pure systems are already nearly equal the cross interaction potential shouldn’t be that crucial.
9 Mixtures

9.2 Hydrogen/Deuterium Mixtures at 20.4 K and 1.5 atm

The simulations of the mixtures at state point A have been started like the ones for the pure systems (see section 8.2). The maximum displacement was set to 0.04 nm and the maximum volume change to 0.06 as in the simulations of the pure systems. The probability for a volume change was again set to 0.01. Apart from the particle and volume moves it is convenient to introduce another type of move for simulations of mixtures: the swapping of particles of different species (see chapter 6 and 7). This move consists of exchanging the center of mass for the two ring-polymers while the relative distance of the beads to the center of mass is kept constant. Such a particle swap is in fact a kind of particle move where two particles are moved simultaneously to the former position of the other particle. The probability for this kind of move was kept constant at 0.1 (i.e., approximately every tenth particle move was replaced by a particle swap). As for the other simulations in physico-chemical units the production phases consisted of 20 blocks of $10 \times 10^4$ passes each.

Tables 9.1 and 9.2 show the results of the simulations at state point A for Trotter numbers of $P = 4$ and $P = 64$, respectively. The results of the simulations with a Trotter number $P = 128$ are given in table 9.3.

The first thing to note is that the acceptance ratios for the particle moves “accrp” don’t depend strongly on the mole fraction of hydrogen (including the extreme values 0% and 100%). The same holds for the acceptance ratio “accrv” of the volume moves. The acceptance ratio “accrs” for the swapping of particles is also quite independent of the concentration of hydrogen and lies around 20%. Since the radius of gyration is larger for hydrogen it is difficult to replace a D$_2$ molecule by a H$_2$ molecule. The other properties change almost linearly between the two pure phases.

Of interest is also how the total potential energy of the mixtures is distributed on the three interactions involved. Since hydrogen and deuterium have different masses the interactions between particles of one species are not symmetrical with respect to the mole fraction of hydrogen. This mass effect can be seen directly in the H$_2$/D$_2$ mixtures.

For $P = 4$ and $P = 64$ the simulations of the 50% mixtures of H$_2$/D$_2$ can be compared to the 50% mixtures of H$_2$/D$_2^\ast$. In the latter mixtures all interaction potentials were equal to the H$_2$ ↔ H$_2$ interaction. Though the interaction potentials in both mixtures are very similar there are considerable differences in the simulation results.

Table 9.4 shows the properties of ideal hydrogen/deuterium mixtures. These ideal values are obtained by linear interpolation between the values of the pure systems. Table 9.5 shows the excess properties of the mixtures at state point A. The excess property $O_{ex}$ is defined by

$$O_{ex} = O_m - O_{id},$$ (9.2.1)

where $O_m$ is the measured value and $O_{id}$ is the ideal value obtained by linear interpolation. For the “more classical” simulations with $P = 4$ the excess properties are very small indicating nearly ideal mixtures. For $P = 64$ and $P = 128$ the excess properties are much larger. This indicates that the deviations from the ideal mixtures are mostly due to quantum mechanics.

---

1This is true for the properties considered in the present work. The ideal values of, e.g., the free enthalpy $G_{id}$ or the entropy $S_{id}$ include additional terms.
Table 9.1: Results for mixtures of $\text{H}_2$ and $\text{D}_2$ at $p = 1.5 \text{ atm}$ and $T = 20.4 \text{ K}$ with a Trotter number $P = 4$. Each simulation consisted of a production phase of 20 blocks of $10 \times 10,000$ passes each. The last line is for a 50\% mixture of $\text{H}_2$ and $\text{D}_2^*$ instead of $\text{D}_2$.

<table>
<thead>
<tr>
<th>$\chi_{\text{H}_2}$</th>
<th>$\text{acrp}$</th>
<th>$\text{acrp}$</th>
<th>$\text{acrv}$</th>
<th>$\text{acrs}$</th>
<th>$\sqrt{\langle r^2 \rangle}$</th>
<th>$\sqrt{\langle r^2 \rangle}$</th>
<th>$\langle \rho \rangle$</th>
<th>$\langle K \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(H$_2$)</td>
<td>(D$_2$)</td>
<td></td>
<td></td>
<td>(H$_2$)</td>
<td>(D$_2$)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>[%]</td>
<td>[%]</td>
<td>[%]</td>
<td>[%]</td>
<td>[%]</td>
<td>[pm]</td>
<td>[pm]</td>
<td>[mol/l]</td>
<td>[J/mol]</td>
</tr>
<tr>
<td>0</td>
<td>27.91(2)</td>
<td>55.90(4)</td>
<td>31.920(4)</td>
<td>48.80(2)</td>
<td>446.1(1)</td>
<td></td>
<td>46.95(2)</td>
<td>469.5(2)</td>
</tr>
<tr>
<td>30</td>
<td>16.63(2)</td>
<td>27.90(2)</td>
<td>56.45(3)</td>
<td>18.07(1)</td>
<td>41.65(1)</td>
<td>32.319(4)</td>
<td>47.51(2)</td>
<td>460.5(2)</td>
</tr>
<tr>
<td>50</td>
<td>16.72(2)</td>
<td>27.83(2)</td>
<td>56.69(4)</td>
<td>17.11(1)</td>
<td>42.18(1)</td>
<td>32.581(5)</td>
<td>46.68(2)</td>
<td>469.5(2)</td>
</tr>
<tr>
<td>70</td>
<td>16.82(2)</td>
<td>27.83(2)</td>
<td>56.96(4)</td>
<td>16.32(9)</td>
<td>42.676(8)</td>
<td>32.832(5)</td>
<td>45.84(2)</td>
<td>476.7(2)</td>
</tr>
<tr>
<td>100</td>
<td>16.92(2)</td>
<td>57.46(5)</td>
<td>43.42(1)</td>
<td>44.59(2)</td>
<td>486.8(3)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>16.08(1)</td>
<td>26.77(1)</td>
<td>56.13(3)</td>
<td>16.80(1)</td>
<td>41.947(8)</td>
<td>32.424(4)</td>
<td>46.92(1)</td>
<td>474.7(1)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\chi_{\text{H}_2}$</th>
<th>$\langle U \rangle$</th>
<th>$\langle U \rangle$</th>
<th>$\langle U \rangle$</th>
<th>$\langle H \rangle$</th>
<th>$\langle V \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(H$_2$)</td>
<td>(H$_2 \leftrightarrow$ D$_2$)</td>
<td>(D$_2$)</td>
<td>(total)</td>
<td></td>
</tr>
<tr>
<td>[%]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[mJ/mol]</td>
</tr>
<tr>
<td>0</td>
<td>-1.5367(8)</td>
<td>-1.0875(7)</td>
<td>20.501(8)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>-0.13275(9)</td>
<td>-0.6365(4)</td>
<td>-0.7400(4)</td>
<td>-1.5092(9)</td>
<td>-1.0455(7)</td>
</tr>
<tr>
<td>50</td>
<td>-0.3755(2)</td>
<td>-0.7503(4)</td>
<td>-0.3661(2)</td>
<td>-1.4918(8)</td>
<td>-1.0191(7)</td>
</tr>
<tr>
<td>70</td>
<td>-0.7140(4)</td>
<td>-0.6259(4)</td>
<td>-0.13318(8)</td>
<td>-1.4730(8)</td>
<td>-0.9930(7)</td>
</tr>
<tr>
<td>100</td>
<td>-1.4458(9)</td>
<td>-0.9556(7)</td>
<td>22.44(1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>-0.3775(1)</td>
<td>-0.7757(3)</td>
<td>-0.3900(1)</td>
<td>-1.5432(6)</td>
<td>-1.0652(5)</td>
</tr>
</tbody>
</table>
Table 9.2: Results for mixtures of H$_2$ and D$_2$ at $p = 1.5$ atm and $T = 20.4$ K with a Trotter number $P = 64$. Each simulation consisted of a production phase of 20 blocks of $10 \times 10^4$ passes each. The last line is for a 50% mixture of H$_2$ and D$_2^*$ instead of D$_2$.

<table>
<thead>
<tr>
<th>$\chi_{H_2}$</th>
<th>accrp</th>
<th>accrp</th>
<th>accrv</th>
<th>accrs</th>
<th>$\sqrt{\langle r^2 \rangle}$</th>
<th>$\sqrt{\langle r^2 \rangle}$</th>
<th>$\langle \rho \rangle$</th>
<th>$\langle K \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(H$_2$)</td>
<td>(D$_2$)</td>
<td>(H$_2$)</td>
<td>(D$_2$)</td>
<td>[pm]</td>
<td>[pm]</td>
<td>[mol/l]</td>
<td>[J/mol]</td>
</tr>
<tr>
<td>0</td>
<td>35.08(2)</td>
<td>49.08(3)</td>
<td>47.561(8)</td>
<td>35.475(3)</td>
<td>44.89(2)</td>
<td>525.4(4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>23.90(3)</td>
<td>36.21(2)</td>
<td>49.91(4)</td>
<td>22.50(1)</td>
<td>35.699(2)</td>
<td>40.71(2)</td>
<td>565.8(4)</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>24.86(2)</td>
<td>37.01(2)</td>
<td>50.53(4)</td>
<td>22.21(2)</td>
<td>35.904(3)</td>
<td>39.24(2)</td>
<td>576.5(5)</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>25.66(3)</td>
<td>37.69(2)</td>
<td>51.04(3)</td>
<td>21.98(1)</td>
<td>48.455(6)</td>
<td>36.93(2)</td>
<td>588.0(6)</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>27.07(2)</td>
<td>51.87(4)</td>
<td>49.091(5)</td>
<td>36.93(2)</td>
<td>588.0(6)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>23.94(3)</td>
<td>35.69(2)</td>
<td>50.07(2)</td>
<td>21.84(1)</td>
<td>47.863(7)</td>
<td>41.08(2)</td>
<td>576.1(3)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\chi_{H_2}$</th>
<th>$\langle U \rangle$</th>
<th>$\langle U \rangle$</th>
<th>$\langle U \rangle$</th>
<th>$\langle H \rangle$</th>
<th>$\langle V \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(H$_2$)</td>
<td>(H$_2$ ↔ D$_2$)</td>
<td>(D$_2$)</td>
<td>(total)</td>
<td>[kJ/mol]</td>
</tr>
<tr>
<td>0</td>
<td>-1.3720(7)</td>
<td>-0.8432(7)</td>
<td>22.289(9)</td>
<td>22.289(9)</td>
<td>-0.8432(7)</td>
</tr>
<tr>
<td>30</td>
<td>-0.11077(6)</td>
<td>-0.5401(3)</td>
<td>-0.6431(3)</td>
<td>-1.2940(7)</td>
<td>-0.7386(8)</td>
</tr>
<tr>
<td>50</td>
<td>-0.3061(2)</td>
<td>-0.6229(3)</td>
<td>-0.3122(2)</td>
<td>-1.2412(7)</td>
<td>-0.6717(7)</td>
</tr>
<tr>
<td>70</td>
<td>-0.5719(4)</td>
<td>-0.5114(3)</td>
<td>-0.11178(7)</td>
<td>-1.1951(8)</td>
<td>-0.6147(8)</td>
</tr>
<tr>
<td>100</td>
<td>-1.1220(7)</td>
<td>-0.5298(6)</td>
<td>27.11(1)</td>
<td>27.11(1)</td>
<td>-0.7104(7)</td>
</tr>
<tr>
<td>50</td>
<td>-0.3093(2)</td>
<td>-0.6469(4)</td>
<td>-0.3340(2)</td>
<td>-1.2902(8)</td>
<td>-0.7104(7)</td>
</tr>
</tbody>
</table>
Table 9.3: Results for 50% mixtures of H$_2$ and D$_2$ at $p = 1.5$ atm and $T = 20.4$ K with a Trotter number $P = 128$.
Each simulation consisted of a production phase of 20 blocks of $10 \times 10\ 000$ passes each.

<table>
<thead>
<tr>
<th>$\chi_{H_2}$</th>
<th>accrp (H$_2$)</th>
<th>accrp (D$_2$)</th>
<th>accrv</th>
<th>accrs</th>
<th>$\sqrt{\langle r^2 \rangle}$ (H$_2$)</th>
<th>$\sqrt{\langle r^2 \rangle}$ (D$_2$)</th>
<th>$\langle \rho \rangle$</th>
<th>$\langle K \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[%]</td>
<td>[%]</td>
<td>[%]</td>
<td>[%]</td>
<td>[pm]</td>
<td>[pm]</td>
<td>[mol/l]</td>
<td>[J/mol]</td>
</tr>
<tr>
<td>0</td>
<td>35.18(2)</td>
<td>42.44(4)</td>
<td></td>
<td></td>
<td>35.13(2)</td>
<td>44.86(2)</td>
<td>526.8(4)</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>24.95(3)</td>
<td>37.09(2)</td>
<td>43.31(4)</td>
<td>22.28(2)</td>
<td>48.063(7)</td>
<td>35.712(4)</td>
<td>40.67(2)</td>
<td>566.5(6)</td>
</tr>
<tr>
<td>100</td>
<td>27.29(2)</td>
<td>44.19(4)</td>
<td></td>
<td></td>
<td>49.128(6)</td>
<td>36.81(2)</td>
<td>588.4(6)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\chi_{H_2}$</th>
<th>$\langle U \rangle$ (H$_2$)</th>
<th>$\langle U \rangle$ (H$_2$ $\leftrightarrow$ D$_2$)</th>
<th>$\langle U \rangle$ (D$_2$)</th>
<th>$\langle U \rangle$ (total)</th>
<th>$\langle H \rangle$</th>
<th>$\langle V \rangle$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
</tr>
<tr>
<td>0</td>
<td>-1.3709(7)</td>
<td>-0.8407(8)</td>
<td>22.30(1)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>-0.3057(2)</td>
<td>-0.6221(4)</td>
<td>-0.3118(2)</td>
<td>-1.2396(8)</td>
<td>-0.669(1)</td>
<td>24.60(1)</td>
</tr>
<tr>
<td>100</td>
<td>-1.1172(7)</td>
<td>-0.5247(6)</td>
<td>27.19(1)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 9.4: Properties of ideal hydrogen/deuterium mixtures at $p = 1.5\,\text{atm}$ and $T = 20.4\,\text{K}$ after tables 9.1 to 9.3. The last two lines are for mixtures with $\text{D}_2^*$ instead of $\text{D}_2$.

<table>
<thead>
<tr>
<th>$\chi_{\text{H}_2}$</th>
<th>$P$</th>
<th>$\rho_{\text{id}}$</th>
<th>$K_{\text{id}}$</th>
<th>$U_{\text{id}}$</th>
<th>$H_{\text{id}}$</th>
<th>$V_{\text{id}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>(%), [mol/l]</td>
<td>[kPa]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[kJ/mol]</td>
<td>[ml/mol]</td>
</tr>
<tr>
<td>30</td>
<td>4</td>
<td>47.55(2)</td>
<td>458.2(1)</td>
<td>-1.5098(6)</td>
<td>-1.0484(5)</td>
<td>21.076(6)</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>42.53(2)</td>
<td>543.9(3)</td>
<td>-1.2979(5)</td>
<td>-0.7503(5)</td>
<td>23.717(7)</td>
</tr>
<tr>
<td>50</td>
<td>4</td>
<td>46.70(1)</td>
<td>466.5(2)</td>
<td>-1.4913(6)</td>
<td>-1.0216(5)</td>
<td>21.471(6)</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>40.91(1)</td>
<td>556.7(4)</td>
<td>-1.2470(5)</td>
<td>-0.6865(5)</td>
<td>24.700(7)</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>40.84(1)</td>
<td>557.6(4)</td>
<td>-1.2441(5)</td>
<td>-0.6827(5)</td>
<td>24.745(7)</td>
</tr>
<tr>
<td>70</td>
<td>4</td>
<td>45.88(2)</td>
<td>474.4(2)</td>
<td>-1.4736(7)</td>
<td>-0.9959(5)</td>
<td>21.848(7)</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>39.36(2)</td>
<td>568.9(4)</td>
<td>-1.1984(5)</td>
<td>-0.6256(5)</td>
<td>25.637(7)</td>
</tr>
<tr>
<td>50</td>
<td>4</td>
<td>46.94(1)</td>
<td>471.4(2)</td>
<td>-1.5447(6)</td>
<td>-1.0701(5)</td>
<td>21.371(6)</td>
</tr>
<tr>
<td></td>
<td>64</td>
<td>41.17(1)</td>
<td>565.2(4)</td>
<td>-1.2961(5)</td>
<td>-0.7272(5)</td>
<td>24.573(7)</td>
</tr>
</tbody>
</table>

This is also supported by the $\text{H}_2/\text{D}_2^*$ mixtures where only the mass difference between the two constituents of the mixtures plays an important role. For mixtures with 30% and 70% hydrogen the simulation results are compared to the measurements of Lambert [53]. The agreement between simulation and measurement is surprisingly good.

Figure 9.1 shows how the excess volumes depend on the mole fraction of hydrogen. The results for the simulations with higher Trotter numbers are better approximations of the measurements by Lambert [53] than the results of $P = 4$. This can be seen even better from figure 9.2, where the excess enthalpies are plotted against the hydrogen concentration. It is interesting to note that the results for the excess enthalpies are in better comparison to the experimental values than the excess volumes. Unfortunately, only very few excess volumes have been measured by Lambert [53] and no errors have been provided.

Figure 9.3 shows the pair-correlation functions of the three different pairings $\text{H}_2 \leftrightarrow \text{H}_2$, $\text{H}_2 \leftrightarrow \text{D}_2$ and $\text{D}_2 \leftrightarrow \text{D}_2$ for a 50% mixture of $\text{H}_2$ and $\text{D}_2$. For the pairings $\text{H}_2 \leftrightarrow \text{H}_2$ and $\text{D}_2 \leftrightarrow \text{D}_2$ the radial distribution functions are almost equal to the pair-correlation functions of the pure systems. As expected, the pair-correlation function for the pairs $\text{H}_2 \leftrightarrow \text{D}_2$ lies between the pure phases. The pair-correlation functions for the other concentrations of hydrogen are similar.
**Table 9.5**: Excess properties of hydrogen/deuterium mixtures at $p = 1.5$ atm and $T = 20.4$ K after tables 9.1 to 9.4. The last two lines are for mixtures with $D_2$ instead of $D_2$. Lines with “exp” as the Trotter number are measurements of Lambert [53].

<table>
<thead>
<tr>
<th>$\chi_{H_2}$</th>
<th>$P$</th>
<th>$\rho_{ex}$ [mol/l]</th>
<th>$K_{ex}$ [J/mol]</th>
<th>$U_{ex}$ [J/mol]</th>
<th>$H_{ex}$ [J/mol]</th>
<th>$V_{ex}$ [ml/mol]</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>30</td>
<td>-0.04(3)</td>
<td>2.3(2)</td>
<td>1.9(1)</td>
<td>2.9(9)</td>
<td>-0.02(1)</td>
</tr>
<tr>
<td></td>
<td>exp</td>
<td>-0.14(3)</td>
<td>7.8(6)</td>
<td>3.9(9)</td>
<td>11.7(9)</td>
<td>10.7</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>-0.02(2)</td>
<td>3.0(3)</td>
<td>-1(1)</td>
<td>2.5(9)</td>
<td>-0.04(1)</td>
</tr>
<tr>
<td></td>
<td>128</td>
<td>-0.20(2)</td>
<td>9.1(6)</td>
<td>5.8(9)</td>
<td>14.8(9)</td>
<td>-0.11(1)</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>-0.17(2)</td>
<td>8.9(7)</td>
<td>4.5(9)</td>
<td>14(1)</td>
<td>-0.15(1)</td>
</tr>
<tr>
<td></td>
<td>exp</td>
<td>-0.04(3)</td>
<td>2.3(3)</td>
<td>1.1(1)</td>
<td>2.9(9)</td>
<td>-0.02(1)</td>
</tr>
<tr>
<td></td>
<td>70</td>
<td>-0.12(3)</td>
<td>7.6(6)</td>
<td>3.3(9)</td>
<td>10.9(9)</td>
<td>-0.14(1)</td>
</tr>
<tr>
<td></td>
<td>exp</td>
<td>-0.02(1)</td>
<td>3.3(2)</td>
<td>1.5(8)</td>
<td>4.9(7)</td>
<td>-0.048(9)</td>
</tr>
<tr>
<td>50</td>
<td>64</td>
<td>-0.09(2)</td>
<td>10.9(5)</td>
<td>5.9(9)</td>
<td>16.8(9)</td>
<td>-0.21(1)</td>
</tr>
</tbody>
</table>
Figure 9.1: Excess volumes for various H$_2$/D$_2$ mixtures at $p = 1.5$ atm and $T = 20.4$ K for various Trotter numbers. The curve “exp” are measurements of Lambert [53] (the line is a guide for the eye). The simulations consisted of $N = 108$ quantum particles and $P = 4, 64$ and 128 beads per particle.
9.2 Hydrogen/Deuterium Mixtures at 20.4 K and 1.5 atm

Figure 9.2: Excess enthalpies for various H$_2$/D$_2$ mixtures at $p = 1.5$ atm and $T = 20.4$ K for various Trotter numbers. The curve “exp” are measurements of Lambert [53] (the line is a guide for the eye). The simulations consisted of $N = 108$ quantum particles and $P = 4, 64$ and 128 beads per particle.
Figure 9.3: Pair-correlation functions for a 50% mixture of H$_2$ and D$_2$ at $p = 1.5$ atm and $T = 20.4$ K. The simulations consisted of $N = 108$ quantum particles and $P = 64$ beads per particle.
9.3 Hydrogen/Deuterium Mixtures at 70.4 K and 747.3 atm

At state point B only 50% mixtures of hydrogen and deuterium have been simulated. The results for $P = 1, 4$ and 64 are shown in Table 9.6. The number of quantum particles was always $N = 108$.

Table 9.7 lists the mixture properties of ideal mixtures of hydrogen and deuterium. The excess properties for the mixtures at state point B are shown in Table 9.8. As at state point A, the excess properties are obviously mostly due to quantum mechanics. Nevertheless, since state B is more classical than state A, the excess values are only roughly doubled between $P = 4$ and $P = 64$. For $P = 1$ the excess values can be neglected.

Figure 9.4 shows the three pair-correlation functions of the mixtures with $N = 108$ quantum particles and $P = 64$ beads per particle.

![Graph of pair-correlation functions](image)

**Figure 9.4:** Pair-correlation functions for a 50% mixture of $H_2$ and $D_2$ at $p = 747.3$ atm and $T = 70.4$ K. The simulations consisted of $N = 108$ quantum particles and $P = 64$ beads per particle.
Table 9.6: Results for mixtures of H\textsubscript{2} and D\textsubscript{2} at $p = 747.3$ atm and $T = 70.4$ K with Trotter numbers $P = 1$, 4 and 64.

Each simulation consisted of a production phase of 20 blocks of $10 \times 10^4$ passes each.

<table>
<thead>
<tr>
<th>(P)</th>
<th>(\chi_{H_2})</th>
<th>(\chi_{accp})</th>
<th>(\chi_{accr})</th>
<th>(\sqrt{\langle r^2 \rangle})</th>
<th>(\langle \rho \rangle)</th>
<th>(\langle K \rangle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>44.906(6)</td>
<td>50.77(4)</td>
<td>96.871(4)</td>
<td>0</td>
<td>48.532(6)</td>
<td>0.878000(1)</td>
</tr>
<tr>
<td>1</td>
<td>44.344(8)</td>
<td>50.37(3)</td>
<td>66.35(1)</td>
<td>26.732(3)</td>
<td>45.598(4)</td>
<td>1.0378(3)</td>
</tr>
<tr>
<td>100</td>
<td>44.032(8)</td>
<td>50.09(3)</td>
<td>26.819(1)</td>
<td>43.621(5)</td>
<td>1.1315(3)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>44.925(9)</td>
<td>36.12(4)</td>
<td>20.2532(6)</td>
<td>45.108(6)</td>
<td>1.071(1)</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>42.621(1)</td>
<td>36.18(3)</td>
<td>70.123(7)</td>
<td>28.190(1)</td>
<td>43.762(7)</td>
<td>1.147(1)</td>
</tr>
<tr>
<td>100</td>
<td>42.615(9)</td>
<td>36.21(5)</td>
<td>28.244(1)</td>
<td>42.569(9)</td>
<td>1.214(1)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>(P)</th>
<th>(\chi_{H_2})</th>
<th>(\langle U \rangle)</th>
<th>(\langle H \rangle)</th>
<th>(\langle V \rangle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>44.906(6)</td>
<td>-1.3306(2)</td>
<td>1.1090(4)</td>
<td>20.622(2)</td>
</tr>
<tr>
<td>1</td>
<td>44.344(8)</td>
<td>-1.3707(3)</td>
<td>1.0701(5)</td>
<td>20.639(3)</td>
</tr>
<tr>
<td>100</td>
<td>44.032(8)</td>
<td>-1.4123(3)</td>
<td>1.0288(5)</td>
<td>20.644(3)</td>
</tr>
<tr>
<td>4</td>
<td>44.925(9)</td>
<td>-1.1960(2)</td>
<td>1.5037(4)</td>
<td>21.949(2)</td>
</tr>
<tr>
<td>64</td>
<td>42.621(1)</td>
<td>-1.28706(8)</td>
<td>-0.5793(1)</td>
<td>-0.28316(7)</td>
</tr>
<tr>
<td>100</td>
<td>42.615(9)</td>
<td>-1.1295(3)</td>
<td>1.865(1)</td>
<td>23.511(5)</td>
</tr>
</tbody>
</table>
Table 9.7: Properties of ideal 50\% hydrogen/deuterium mixtures at \( p = 747.3 \text{ atm} \) and \( T = 70.4 \text{ K} \) after table 9.6

<table>
<thead>
<tr>
<th>( P ) [mol/l]</th>
<th>( \rho_{\text{id}} ) [kJ/mol]</th>
<th>( K_{\text{id}} ) [kJ/mol]</th>
<th>( U_{\text{id}} ) [kJ/mol]</th>
<th>( H_{\text{id}} ) [kJ/mol]</th>
<th>( V_{\text{id}} ) [ml/mol]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>48.506(5)</td>
<td>0.878000(1)</td>
<td>-1.3715(2)</td>
<td>1.0689(3)</td>
<td>20.633(2)</td>
</tr>
<tr>
<td>4</td>
<td>44.610(3)</td>
<td>1.0847(2)</td>
<td>-1.1882(1)</td>
<td>1.5961(3)</td>
<td>22.447(2)</td>
</tr>
<tr>
<td>64</td>
<td>43.839(5)</td>
<td>1.1425(7)</td>
<td>-1.1518(2)</td>
<td>1.7210(7)</td>
<td>22.850(3)</td>
</tr>
</tbody>
</table>

Table 9.8: Excess properties of 50\% hydrogen/deuterium mixtures at \( p = 747.3 \text{ atm} \) and \( T = 70.4 \text{ K} \) after tables 9.6 and 9.7

<table>
<thead>
<tr>
<th>( P ) [mol/l]</th>
<th>( \rho_{\text{ex}} ) [kJ/mol]</th>
<th>( K_{\text{ex}} ) [kJ/mol]</th>
<th>( U_{\text{ex}} ) [kJ/mol]</th>
<th>( H_{\text{ex}} ) [kJ/mol]</th>
<th>( V_{\text{ex}} ) [ml/mol]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.015(9)</td>
<td>0</td>
<td>0.8(4)</td>
<td>1.2(6)</td>
<td>0.006(4)</td>
</tr>
<tr>
<td>4</td>
<td>-0.043(5)</td>
<td>2.3(4)</td>
<td>1.2(1)</td>
<td>4.4(5)</td>
<td>0.010(3)</td>
</tr>
<tr>
<td>64</td>
<td>-0.077(9)</td>
<td>4.5(1)</td>
<td>2.3(4)</td>
<td>8(1)</td>
<td>0.020(5)</td>
</tr>
</tbody>
</table>
10 Conclusions

Pure molecular hydrogen and deuterium have been simulated with the Path-Integral Monte Carlo method in the isothermal-isobaric ensemble at 20.4 K, 1.5 atm and 70.4 K, 747.3 atm. First the number of “quantum” particles was held constant at \( N = 108 \) and the Trotter number was varied between \( P = 1 \) and \( P = 64 \) (\( P = 128 \) for some systems). At both state points considerable differences between classical simulations (Trotter number \( P = 1 \)) and quantum mechanical simulations with \( P = 64 \) were observed, though these differences were larger for state point A than for state point B since the latter is the “more classical” one. For Trotter numbers \( 1 < P < 64 \) it was possible to observe how the calculated properties change when quantum mechanics is “turned on”: For \( P \leq 16 \) the values changed rapidly with the Trotter number while for \( P > 16 \) the properties began to converge. In a second step the Trotter number was held constant at \( P = 64 \) and the variations of the system properties with the number of quantum particles was studied. From both types of simulations it was concluded that simulations with \( N = 108 \) quantum particles and \( P = 64 \) beads per particle are good approximations when time and memory considerations are taken into account (of course, larger \( N \) and \( P \) are always preferable if possible).

After the pure phases mixtures have been studied with the same Path-Integral Monte Carlo method. Mixtures with mole fractions of 30, 50 and 70 % hydrogen were simulated (in addition to the pure phases with 0 and 100 % hydrogen). The particle number was held constant at \( N = 108 \) and the Trotter number was varied from 1 up to 128. From the simulations of the pure systems and of the mixtures the excess values for the enthalpy and the volume were calculated. Additionally, the excess values of the density, the kinetic energy and the potential energy were also calculated. As the results showed the excess properties also depend strongly on the Trotter number. From this it can be concluded that the deviations from an ideal mixture are mainly due to the effects of quantum mechanics. For mixtures with 30 and 70 % hydrogen at \( T = 20.4 \) K and \( p = 1.5 \) atm the simulation results with \( P = 64 \) were compared with measurements after Lambert [53]. The agreement is surprisingly good.

To further support the conclusion that the excess properties are mainly due to quantum effects one should study systems with higher Trotter numbers. Also more realistic (though at first still spherically symmetric) potentials could be used (especially for the hydrogen-deuterium interaction). In general, it would be advisable to measure and simulate more mixtures with different mole fractions of hydrogen. Another thing of interest would be to replace the primitive method for splitting the density matrix by methods that are at least of the order \( 1/P^4 \).
Appendix
A Mathematical Addendum

This appendix gives a brief collection of mathematical results used throughout the preceding chapters. Much of the material is taken from Bröcker [13], Fleming [28] and Straumann [86, 87].

A.1 Convex Functions

This section gives a short introduction on convex sets and convex functions. The omitted proofs can be found in Barner and Flohr [9], Bröcker [13], Fleming [28] and Heuser [38, 39].

Definition A.1.1. $K \subseteq \mathbb{R}^n$ is called convex :\iff $\forall x_1, x_2 \in K$ and $\forall \lambda \in [0,1]$ follows $x(\lambda) := x_1 + \lambda(x_2 - x_1) = (1 - \lambda)x_1 + \lambda x_2$ is also in $K$. I.e., $K$ is convex if and only if for each two points $x_1, x_2 \in K$ their connecting line segment is entirely in $K$.

Example A.1.2. The trivial convex subsets of $\mathbb{R}^n$ are the empty set, all subsets of $\mathbb{R}^n$ with only one element and $\mathbb{R}^n$ itself. All intervals $I \subseteq \mathbb{R}$ are also convex sets.

Definition A.1.3. Let $K \subseteq \mathbb{R}^n$ be a convex set. $f : K \to \mathbb{R}$ is called convex on $K$ :\iff for each two distinct points $x_1, x_2 \in K$ and $\forall \lambda \in (0,1)$ follows

$$f((1 - \lambda)x_1 + \lambda x_2) \leq (1 - \lambda)f(x_1) + \lambda f(x_2). \quad (A.1.1)$$

$f$ is called concave, if $-f$ is convex. If the $\leq$ in (A.1.1) is replaced by $<$ or $>$, $f$ is called strictly convex or strictly concave.

Corollary A.1.4. Let $f : I \subseteq \mathbb{R} \to \mathbb{R}$ be differentiable on the open interval $I$. Then $f$ is (strictly) convex or concave on $I$, if and only if $f'$ is (strictly) increasing or decreasing on $I$.

Lemma A.1.5. Let $f : I \subseteq \mathbb{R} \to \mathbb{R}$ be differentiable on the open interval $I$. For every fixed $x_0 \in I$ the tangent $T_{|x_0}$ to $f$ in the point $(x_0, f(x_0))$ is defined as $T_{|x_0}^f (x) := f(x_0) + f'(x_0)(x-x_0)$ (see Fig. A.1(a)). If $f$ is convex or concave on $I$, then $f(x) \geq T_{|x_0}^f (x)$ or $f(x) \leq T_{|x_0}^f (x)$ $\forall x \in I$. In the strict cases $f(x) > T_{|x_0}^f (x)$, where $x \neq x_0$.

Proof. For $x = x_0$ nothing has to be proved. Let $x > x_0$. From the mean value theorem follows $\exists \xi \in [x_0, x] : f'\xi(x) = (f(x) - f(x_0))/(x - x_0)$. Since $f$ is convex, from Corollary A.1.4 follows $f'(\xi) \geq f'(x_0)$. Hence $f(x) \geq f(x_0) + f'(x_0)(x-x_0) = T_{|x_0}^f (x)$. For $x < x_0 \exists \xi \in [x, x_0] : f'\xi(x) = (f(x_0) - f(x))/(x_0 - x)$. Since $f'$ is increasing, $f'\xi \leq f'(x_0)$. Hence $f(x) \geq f(x_0) + f'(x_0)(x-x_0) = T_{|x_0}^f (x)$. The proof for concave $f$ and for the strict cases is analogous.

Corollary A.1.6. Let $f : I \subseteq \mathbb{R} \to \mathbb{R}$ be twice differentiable on the open interval $I$. Then $f$ is convex or concave on $I$, if and only if $f'' \geq 0$ or $f'' \leq 0$ on $I$. $f$ is strictly convex or strictly concave on $I$, if and only if $f'' > 0$ or $f'' < 0$ on $I$. 

97
A Mathematical Addendum

A.2 Legendre Transformations

Legendre transformations are used, e.g. in Hamiltonian mechanics, in thermodynamics, in the theory of ordinary differential equations (e.g. Baule [10, §10]) or in variational calculus (e.g. Courant and Hilbert [22, §IV.9]).

The main idea of Legendre transformations is the duality of a graph of a function \( f \) seen as a set of points or seen as the envelope of its tangents. Since both views describe the same function they contain the same information about the function.

Let \( p = f'(x_0) \) be the slope and \( g \) be the negative \( y \)-intercept of the tangent \( T \big|_{x_0} \) to \( f \) at the point \((x_0, f(x_0))\) (see Fig. A.1(a)). Then \( T \big|_{x_0} (x) = f(x_0) + p(x - x_0) \) and hence the negative \( y \)-intercept \( g = -T \big|_{x_0} (0) \) is given by \( g = px_0 - f(x_0) \). Now this can be repeated for all \( x_0 \) in the domain of \( f \), resulting in a set of pairs \((x_0, f(x_0))\) and a set of pairs \((p, g)\).

One of these pairs \((p, g)\) by itself doesn’t tell us anything about the function \( f \). However, the values \((p, g)\) for every \( p \) define a new function \( g = g(p) \), which is called the Legendre transformation of \( f \). This function returns for a given slope \( p \) the negative \( y \)-intercept of a straight line. Thus, as \((x, f(x))\) defines a set of points, \((p, g(p))\) defines a set of straight lines, which—by construction—are all tangents to \( f \). Hence the envelope of these lines is \( f \) again.

\[ g(p) = px_0 - f(x_0) \]

A problem occurs, if for some \( x_1 \neq x_2 \) one has \( p = f'(x_1) = f'(x_2) \) but \( f(x_1) - px_1 \neq f(x_2) - px_2 \). In this case the slope \( p \) is assigned two different \( y \)-intercepts and hence \( g \) is no function of \( p \). To avoid this problem, one usually assumes \( f \) to be strictly convex.

Another way of defining \( g(p) \) is as follows (see Arnold [5, §14]): Let \( p \) be a given number and consider the straight line \( y = px \) (see Fig. A.1(b)). Let \( x = x(p) \) be the abscissa at which

\[ \text{Figure A.1: Definition of the Legendre transformation} \]

A first draft of this section was based on Hillman [40]. Parts are also taken from Arendt [4].
the curve \( y = f(x) \) is farthest from the straight line in the vertical direction: for each \( p \) the function \( F(p, x) = px - f(x) \) has a maximum with respect to \( x \) at \( x = x(p) \). Now the Legendre transformation is defined as \( g(p) = F(p, x(p)) \). The abscissa \( x(p) \) is defined by the extremal condition \( \frac{\partial}{\partial x} F(p, x) = p - f'(x) = 0 \), i.e., \( p = f'(x) \).

If differentiability is dropped and only convexity (but not strict convexity) is retained, then the Legendre transformation can be defined in the following way (see Arendt [4]): For a given slope \( p \) the negative \( y \)-intercept \( g \) of the straight line \( h : x \mapsto px - g \) is chosen such that \( h \) lies below the function \( f \) and never intersects it (if this is possible). Then \( g \) is decreased till \( h \) "touches" \( f \), i.e., \( px_0 - g = f(x_0) \) for some \( x_0 \), which might not be unique. Doing this for every slope \( p \) where this makes sense, this gives a function \( g(p) \).

With this definition the Legendre transformation of the straight line \( x \mapsto ax + b \) is given by the single point \((p = a, g(p) = -b)\). Another example is the convex polygon

\[
\begin{align*}
    f : x &\mapsto \begin{cases} 
        a_1x + b_1 & x \leq x_0 \\
        a_2x + b_2 & x \geq x_0 
    \end{cases},
\end{align*}
\]

where \( a_2 > a_1 \) and \( a_1x_0 + b_1 = a_2x_0 + b_2 \). Obviously \((a_1, -b_1)\) and \((a_2, -b_2)\) belong to the Legendre transformation of \( f \). But what happens at \( x_0 \)? According to the last definition, every line \( x \mapsto px - g \) through the point \((x_0, f(x_0))\) touches \( f \) in \( x_0 \) as long as \( a_1 < p < a_2 \) and \( b_1 > -g > b_2 \). Hence the Legendre transformation of \( f \) at \( x_0 \) is the straight line segment connecting \((a_1, -b_1)\) and \((a_2, -b_2)\) with slope \( x_0 \) \((g(p) = px_0 - f(x_0))\).

This is a general feature of Legendre transformations: points and lines are “dual” to one another.

In the following, the Legendre transformation is defined more precisely for differentiable and strictly convex functions.

### A.2.1 Legendre Transformations in Hamiltonian Mechanics

**Definition A.2.1.** Let \( f : K \subseteq \mathbb{R}^n \to \mathbb{R} \) be strictly convex on the convex set \( K \). The Legendre transformation of \( f \) is defined by

\[
    L_M[f](p) := \sup_{x \in K} \{ (p, x) - f(x) \}, \tag{A.2.1a}
\]

where \((\cdot, \cdot)\) denotes the scalar product in \( \mathbb{R}^n \). If \( f \) is strictly concave its Legendre transformation is defined by

\[
    L_M[f](p) := \inf_{x \in K} \{ (p, x) - f(x) \}. \tag{A.2.1b}
\]

**Remark.** From Definition A.2.1 follows for strictly convex or strictly concave \( f \) that \( L_M[f](p) \geq (p, x) - f(x) \) or \( L_M[f](p) \leq (p, x) - f(x) \). This gives Young’s inequalities \((p, x) \leq f(x) + L_M[f](p) \) or \( (p, x) \geq f(x) + L_M[f](p) \).

**Corollary A.2.2.** \( L_M[f + c] = L_M[f] - c \) for every constant \( c \in \mathbb{R} \).
Remark. Definition A.2.1 is very general but not really usable. A more useful definition of $L_M[f]$ for differentiable $f$ is given in the following Lemma.

Lemma A.2.3. Let $f : I \subseteq \mathbb{R} \to \mathbb{R}$ be differentiable and strictly convex or strictly concave on the open interval $I$. Then $L_M[f](p) = xf'(x) - f(x)$, where $x$ is the unique number such that $p = f'(x)$, i.e., $L_M[f] : f'(I) \subseteq \mathbb{R} \to \mathbb{R}$ and $L_M[f](p) = f^{-1}\left((f^{-1}(p))p - f(f^{-1}(p))\right)$.

Remark. Since $f$ is strictly convex, $f'$ is a strictly increasing function, but usually not convex. The strict convexity of $f$ ensures also, that $p = f'(x)$ can be solved for $x$. The same holds for $f$ strictly concave.

Proof. Let $x_0 \in I$ and $p_0 = f'(x_0)$. For strictly convex $f$ one needs to show that $xp_0 - f(x)$ is always less than $x_0p_0 - f(x_0)$. Let $T|_{x_0}$ be the tangent to $f$ in the point $(x_0, f(x_0))$ as defined in Lemma A.1.5 (see Fig. A.1(a)). Write the difference $(x_0p_0 - f(x_0)) - (xp_0 - f(x))$ as $f(x) - f(x_0) - p_0(x - x_0) = f(x) - T|_{x_0}(x)$. Lemma A.1.5 states that this difference is always $> 0 \forall x \neq x_0$. This shows that the maximum value of $xp_0 - f(x)$ as one varies $x$ occurs for $x = x_0$. But $p_0 = f'(x_0)$, so $L_M[f](p_0) = x_0f'(x_0) - f(x_0)$. In the strict concave case Lemma A.1.5 guarantees that $xp_0 - f(x)$ is always greater than $x_0p_0 - f(x_0) \forall x \neq x_0$. \hfill \Box

Remark. $f'(I)$ is not necessarily an open interval in $\mathbb{R}$. After Heuser [38, 37.1 on p. 231] for strictly monotone $f'$, $f'(I)$ is an open interval only for $f$ continuous, hence for $f$ continuously differentiable. Especially, $f'(I)$ is an open interval for $f$ at least twice differentiable.

Corollary A.2.4. Let $f : I \subseteq \mathbb{R} \to \mathbb{R}$ be twice continuously differentiable\(^3\) and strictly convex or strictly concave on the open interval $I$. If $p = f'(x)$ then $x = \left(L_M[f]\right)'(p)$, i.e. $(L_M[f])' : f'(I) \to I$ and $(L_M[f])' = f^{-1}$.

Proof. Since $f$ is strictly convex or strictly concave, $f'' \geq 0$ on $I$. According to the inverse function theorem $\left(f^{-1}\right)'(p) = 1/f''(x)$. With $L_M[f](p) = f^{-1}(p)p - f(f^{-1}(p))$ follows $(L_M[f])'(p) = (1/f''(x))p + f^{-1}(p) - f'(f^{-1}(p))(1/f''(x)) = f^{-1}(p) = x$. \hfill \Box

Corollary A.2.5. Let $f : I \subseteq \mathbb{R} \to \mathbb{R}$ be twice continuously differentiable on the open interval $I$. If $f$ is strictly convex or strictly concave, its Legendre transform is also strictly convex or strictly concave.

Proof. For $f$ strictly convex $f'$ is strictly increasing. After Corollary A.2.4 the inverse of $f'$ is $(L_M[f])'$. Since the inverse of a strictly increasing function is itself strictly increasing $(L_M[f])'$ is strictly increasing and $L_M[f]$ is strictly convex. For $f$ strictly concave the proof is analogous. \hfill \Box

Corollary A.2.6. Let $f : I \subseteq \mathbb{R} \to \mathbb{R}$ be twice continuously differentiable and strictly convex or strictly concave on the interval $I$. Then $L_M[L_M[f]](x) = f(x) \forall x \in I$, i.e. $L_M$ is involutive.

\(^2\)At a first look on $L_M[f](p) = xf'(x) - f(x)$ one would assume that $L_M[cf] = cL_M[f]$ for every constant $c \in \mathbb{R}$ and $L_M[f + g] = L_M[f] + L_M[g]$. But this is not the case, since in general $(cf)^{-1} \neq f^{-1}$ and $(f + g)^{-1} \neq f^{-1} \neq g^{-1}$, respectively.

\(^3\)This is sometimes denoted as $f \in \mathcal{C}^2(I \subseteq \mathbb{R}, \mathbb{R})$. 
A.2 Legendre Transformations

Proof. Let \( x \in I \), let \( g(p) = L_M[f](p) = xf'(x) - f(x) \) with \( p = f'(x) \) and let \( h(y) = L_M[g](y) = pg'(p) - g(p) \) with \( y = g'(p) = x \) (after Corollary A.2.4). Hence \( h(y) = h(x) = f(x) \) and \( L_M[L_M[f]](x) = f(x) \) \( \forall x \in I \). □

The commutative diagram of \( L_M[f] \) for an \( f \in C^2(I \subseteq \mathbb{R}, \mathbb{R}) \) is shown in Fig. A.2(a).

**Examples A.2.7.**

1. \( f(x) = x^α/α \) on \( \mathbb{R}^+ \), \( α > 1 \). Then \( p = f'(x) = x^{α-1} \) and \( g(p) = L_M[f](p) = xf'(x) - f(x) = x^α - x^α/α = x^α(1 - 1/α) = p^β/β \) with \( 1/α + 1/β = 1 \).

   With \( y = g'(p) = p^{β-1} \) follows \( h(y) = L_M[L_M[f]](y) = pg'(p) - g(p) = p^β - p^{β}/β = p^β(1 - 1/β) = y^α/α \). Since \( y = g'(f'(x)) = x^{(α-1)(β-1)} = x \), \( L_M[L_M[f]](x) = f(x) \) \( \forall x \in \mathbb{R}^+ \). Further \( px \leq x^α/α + p^β/β \) for \( x, p \in \mathbb{R}^+ \) and \( 1/α + 1/β = 1 \).

2. \( f(x) = \ln x \) on \( \mathbb{R}^+ \). Then \( p = f'(x) = 1/x \) and \( g(p) = L_M[f](p) = xf'(x) - f(x) = 1 - \ln x = 1 + \ln p \). \( y = g'(p) = 1/p \) gives \( h(y) = L_M[L_M[f]](y) = pg'(p) - g(p) = 1 - (1 + \ln p) = -\ln p = \ln y \). Since \( y = g'(f'(x)) = 1/(1/x) = x \), \( L_M[L_M[f]](x) = f(x) \) \( \forall x \in \mathbb{R}^+ \). Further \( px \geq 1 + \ln x + \ln p \) for \( x, p \in \mathbb{R}^+ \).

3. \( f(x) = x \ln x \) on \( \mathbb{R}^+ \). Then \( p = f'(x) = \ln x + 1 \) and \( g(p) = L_M[f](p) = xf'(x) - f(x) = x \ln x + x - x \ln x = x = \exp(p - 1) \). With \( y = g'(p) = \exp(p - 1) \) follows \( h(y) = L_M[L_M[f]](y) = pg'(p) - g(p) = \exp(p - 1) - \exp(p - 1) = \exp(p - 1)(p - 1) = y \ln y \).

   Since \( y = g'(f'(x)) = \exp(\ln x + 1 - 1) = x \), \( L_M[L_M[f]](x) = f(x) \) \( \forall x \in \mathbb{R}^+ \). Further \( px \leq x \ln x + \exp(p - 1) \) for \( x \in \mathbb{R}^+ \), \( p \in \mathbb{R} \).

![Figure A.2: Commutative diagrams for the Legendre transformation of an f ∈ C^2(I ⊆ R, R): as defined in Hamiltonian mechanics (a); as defined in thermodynamics (b) performing Legendre transformations in thermodynamics.](image)

The Legendre transformation in Hamiltonian mechanics is used to switch back and forth between the Hamilton function (the Hamiltonian) and the Lagrange function.

**A.2.2 Legendre Transformations in Thermodynamics**

**Definition A.2.8.** Let \( f : K \subseteq \mathbb{R}^n \rightarrow \mathbb{R} \) be strictly convex on the convex set \( K \). The (thermodynamical) Legendre transformation of \( f \) is defined by

\[
L_T[f](p) := \inf_{x \in K} \{ f(x) - (p, x) \}.
\]  

(A.2.2a)
If $f$ is strictly concave its Legendre transformation is defined by

$$L_T[f](p) := \sup_{x \in \mathbb{R}} \{ f(x) - (p, x) \}. \quad (A.2.2b)$$

**Remark.** Young’s inequalities become $(p, x) \leq f(x) - L_T[f](p)$ for $f$ strictly convex and $(p, x) \geq f(x) - L_T[f](p)$ for $f$ strictly concave.


**Lemma A.2.9.** Let $f : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$ be differentiable and strictly convex or strictly concave on the open interval $I$. Then $L_T[f](p) = f(x) - x f'(x)$, where $x$ is the unique number such that $p = f'(x)$, i.e. $L_T[f] = (L_T[f])'(p) = x$. For $f$ strictly convex the proof is analogous to the proof of Lemma A.2.3.

**Proof.** Since $L_T[f] = -L_M[f]$, $(L_T[f])'(p) = -(L_T[f])'(p) = -x$. \hfill \Box

**Corollary A.2.10.** Let $f : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$ be twice differentiable and strictly convex or strictly concave on the open interval $I$. If $p = f'(x)$ then $x = -(L_T[f])'(p)$, i.e. $(f')^{-1} = -(L_T[f])'$. For $f$ strictly convex the proof is analogous.

**Proof.** Since $L_T[f] = -L_M[f]$, $(L_T[f])'(p) = -(L_T[f])'(p) = -x$. \hfill \Box

**Corollary A.2.11.** Let $f : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$ be twice differentiable on the open interval $I$. If $f$ is strictly convex or strictly concave, its Legendre transform is strictly convex or strictly convex, respectively.

**Proof.** For $f$ strictly convex $f'$ is strictly increasing. After Corollary A.2.10 the inverse of $f'$ is $-(L_T[f])$. Since the inverse of a strictly increasing function is itself strictly increasing $(L_T[f])$ is strictly decreasing and $L_T[f]$ is strictly concave. For $f$ strictly concave the proof is analogous. \hfill \Box

**Corollary A.2.12.** Let $f : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$ be twice differentiable and strictly convex or strictly concave on the interval $I$. Then $L_T[L_T[f]](x) = f(x) \forall x \in I.$

**Proof.** Analogous as in the proof of Corollary A.2.6 one can show, that $h(y) = f(x)$. But $y = g'(p) = -x$ after Corollary A.2.10. Hence $L_T[L_T[f]](x) = f(x)$. \hfill \Box

The commutative diagram of $L_T[f]$ for an $f \in C^2(I \subseteq \mathbb{R}, \mathbb{R})$ is shown in Fig. A.2(b).

**Examples A.2.13.**

1. $f(x) = x^\alpha/\alpha$ on $\mathbb{R}^+$. Then $p = f'(x) = x^{-\alpha-1}$ and $g(p) = L_T[f](p) = f(x) - x f'(x) = x^\alpha/\alpha - x^\alpha = x^\alpha(1/\alpha-1) = -p^\beta/\beta$ with $1/\alpha+1/\beta = 1$. With $y = g'(p) = -p^\beta-1$ follows $h(y) = L_T[L_T[f]](y) = g(p) - pg'(p) = -p^\beta/\beta + p^\beta = p^\beta/\beta = -y^\alpha/\alpha$. Since $y = g'(f'(x)) = (-x)^{(\alpha-1)(\beta-1)} = -x$, $L_T[L_T[f]](x) = f(x) \forall x \in \mathbb{R}^+$. Further $px \leq x^\alpha/\alpha + p^\beta/\beta$ for $x, p \in \mathbb{R}^+$ and $1/\alpha+1/\beta = 1$.

2. $f(x) = \ln x$ on $\mathbb{R}^+$. Then $p = f'(x) = 1/x$ and $g(p) = L_T[f](p) = f(x) - x f'(x) = \ln x - 1 = -1 + \ln p$. $y = g'(p) = -1/p$ gives $h(y) = L_T[L_T[f]](y) = g(p) - pg'(p) = -1 + \ln p = -1 - \ln y$. Since $y = g'(f'(x)) = -1/(1/x) = -x$, $L_T[L_T[f]](x) = f(x) \forall x \in \mathbb{R}^+$. Further $px \geq 1 + \ln x + \ln p$ for $x, p \in \mathbb{R}^+$.\footnote{Balian [7, p. 255] comments on this: "Apart from a few signs, the Legendre transformation is symmetric."}

A Mathematical Addendum
3. \( f(x) = x \ln x \) on \( \mathbb{R}^+ \). Then \( p = f'(x) = \ln x + 1 \) and \( g(p) = \mathcal{L}_T[f](p) = f(x) - xf'(x) = x \ln x - x \ln x - x = -x = -\exp(p-1) \). With \( y = g'(p) = -\exp(p-1) \) follows \( h(y) = \mathcal{L}_T[\mathcal{L}_T[f]](y) = g(p) - pg'(p) = -\exp(p-1)+p\exp(p-1) = \exp(p-1)(p-1) = -y \ln y \). Since \( y = g'(f'(x)) = -\exp(\ln x+1-1) = -x \), \( \mathcal{L}_T[\mathcal{L}_T[f]](-x) = f(x) \forall x \in \mathbb{R}^+ \). Further \( px \leq x \ln x + \exp(p-1) \) for \( x \in \mathbb{R}^+ \), \( p \in \mathbb{R} \).

**Remark.** The Legendre transformation can be defined under very weak conditions. It is then called the Legendre-Fenchel duality (see Rockafellar [72]).

### A.3 Laplace Transformation

**Definition A.3.1.** Let \( f : [0, \infty) \rightarrow \mathbb{C} \) be a continuous function. If \( |f(x)| \leq \text{const } e^{\alpha x} \) for all \( x \geq 0 \) and some real \( \alpha \), the Laplace transformation of \( f \) is defined by \([33]\)

\[
\mathcal{L}[f](s) := \int_0^\infty e^{-sx} f(x) \, dx ,
\]

where \( s \in \mathbb{C} \) and \( \Re s > \alpha \). The dimensionless (or proper) Laplace transformation of \( f \) is given by

\[
\mathcal{L}_p[f](s) := s \int_0^\infty e^{-sx} f(x) \, dx .
\]

**Remark.** The inverse of the Laplace transformation is given by

\[
f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{(\sigma+i\tau)x} \mathcal{L}[f](\sigma + i\tau) \, d\tau ,
\]

where \( \sigma \) is a real number with \( \sigma > \alpha \).
B The (Pseudo) Random Number Generator

B.1 Uniform Deviates

Since the runtime of the simulations is very long it is desirable to save the current state of the simulations at regular time-intervals. To do this the actual simulation programs were started within a Perl script, read in the current state, calculated the observables for some time, saved the current state and exited to be called again from the Perl script. Hence the current state of the (pseudo) random number generator (RNG) had also to be saved. Unfortunately, the RNG of the standard C++ library has no means for saving and restoring its state. Hence I had to write my “own” RNG, which is based on ran2 from Press et al. [70].

```plaintext
#include <iostream.h>

class float_rng {

  int ntab;
  long idum;
  long idum2;
  long iy;
  long * iv;

public:

  float_rng(const long nseed = 1, const int nntab = 32);
  ~float_rng();

  float operator()();

};
```

#ifndef _FLOAT_RNG
#define _FLOAT_RNG

#include <iostream.h>

class float_rng {

  int ntab;
  long idum;
  long idum2;
  long iy;
  long * iv;

public:

  float_rng(const long nseed = 1, const int nntab = 32);
  ~float_rng();

  float operator()();

};

#endif

// float_rng.h

//Random number generator (RNG) based on ran2 from:
// William H. Press, Saul A. Teukolsky, William T. Vetterling, and
// Cambridge University Press, 1995
//
B The (Pseudo) Random Number Generator

friend ostream& operator<<(ostream & s , const float_rng & Gen);  
friend istream & operator>>(istream & s , float_rng & Gen);
}; // float_rng

#include <iostream>
#include "float_rng.h"

// global constants

const long IM1 = 2147483563;
const long IM2 = 2147483399;
const double AM = 1.0/IM1;
const long IMM1 = IM1−1;
const long IA1 = 40014;
const long IA2 = 40692;
const long IQ1 = 53668;
const long IQ2 = 52774;
const long IR1 = 12211;
const long IR2 = 3791;
const float RNMX = 1.0f−FLT_EPSILON;

float_rng::float_rng(const long nseed , const int nntab) : ntab(nntab)
{
    iv = new long[nntab];

    // seed must be >= 1

    if (nseed >= 1) idum = nseed;
    else idum = 1;

    idum2 = idum;
    for (int j = ntab+7; j >= 0; --j) {
        long k = idum/IQ1;
        idum = IA1*(idum−k*IQ1)−k*IR1;
        if (idum < 0) idum += IM1;
        if (j < ntab) iv[j] = idum;
    }

106
B.2 Normal Deviates

If one can generate random numbers uniformly on \([0,1]\) it is easy to generate from them random numbers that are normally distributed. To draw random numbers from a Gaussian
distribution
\[ w(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(x-\mu)^2/2\sigma^2} \]
with mean \( \mu \) and variance \( \sigma^2 \) one uses the Box-Muller method (see Vesely [91]): Let \( \xi \) and \( \eta \) be two random numbers uniformly distributed on \([0,1)\). Then
\[
\begin{align*}
x &= \mu + \sqrt{-2\sigma^2 \ln(1-\xi)} \cos 2\pi \eta, \\
y &= \mu + \sqrt{-2\sigma^2 \ln(1-\xi)} \sin 2\pi \eta
\end{align*}
\]
are two uncorrelated normal deviates.
C The Code

In the following the simulation programs for the Lennard-Jones mixtures in physico-chemical units are listed. The changes for pure Lennard-Jonesium are obvious, though the mixture programs could also be used for the pure systems (by setting the number of particles of substance A nA to zero).

C.1 Initialization and Re-Initialization

C.1.1 Random Initial Configuration: gnptpimclj

gnptpimclj creates a random (“gas”) initial configuration for NpT Path-Integral Monte Carlo of Lennard-Jones mixtures.

//
// gnptpimclj.cpp
//
// create random ("gas") initial configuration for NpT Path–Integral
// Monte Carlo of Lennard–Jones mixtures
//
// the following units are used:
// Length: 1 nm
// Volume: 1 nm**3
// Density: 1/nm**3
// Energy: 1 kJ/mol
// Pressure: 1 atm
// Temperature: 1 K
//

#include <math.h>
#include <stdlib.h>
#include <iostream.h>
#include <fstream.h>

#include "float_rng.h"

// global data; to be stored in the .dat files

int n; // total number of particles
int nA; // number of particles of substance A
int np; // Trotter number (beads/particle)

float epsA; // LJ–epsilon (in kJ/mol) for substance A
C The Code

float sigA; // LJ-sigma (in nm) for substance A
float mA; // particle mass (in kg) for substance A
float epsB; // LJ-epsilon (in kJ/mol) for substance B
float sigB; // LJ-sigma (in nm) for substance B
float mB; // particle mass (in kg) for substance B

float p; // pressure
float t; // temperature
float v; // volume

float disp; // particle displacement

float dvv; // volume change per volume dv/v
float pvch; // probability for volume change

float pswp; // probability for swapping of particles

float dr; // channel width for pair-correlation

long nt; // "time"
int ntskip; // thinning factor
int ntprint; // print control variables every ntprint*ntskip passes
int ntjob; // ntjob*ntskip passes

float** x; // x-coordinates for the n particles
float** y; // y-coordinates for the n particles
float** z; // z-coordinates for the n particles

float_rng the_rng; // the RNG

// global data; not to be stored in the .dat files

const float au_c = 1.66053873e-27; // atomic mass unit in kg
const float NA_c = 6.02214199e23; // Avogadro's constant in 1/mol
const float R_c = 8.314472e-3; // molar gas constant in kJ/(mol K)

// main

int main() {
    cout << "n: "; cin >> n;
    cout << "nA: "; cin >> nA; cout << "chiA==" << float(nA)/n << endl;
    cout << "np: "; cin >> np;

    if ( (np%2 != 0) && (np != 1) ) {
        cout << "np has to be even or ==1" << endl;
        exit(1);
    }

    // main body
}

110
C.1 Initialization and Re-Initialization

cout << "A: \varepsilon /k \text{[K]}:\_;\ cin >> \varepsilon A; \ \varepsilon A \ast= R_c;\n";
cout << "A:\nu \text{[nm]}:\_;\ cin >> \nu A; \ \nu A \ast= \text{au};\n";
cout << "A:\mu \text{[au]}:\_;\ cin >> \mu A; \ \mu A \ast= \text{au};\n";
cout << "B: \varepsilon /k \text{[K]}:\_;\ cin >> \varepsilon B; \ \varepsilon B \ast= R_c;\n";
cout << "B:\nu \text{[nm]}:\_;\ cin >> \nu B; \ \nu B \ast= \text{au};\n";
cout << "B:\mu \text{[au]}:\_;\ cin >> \mu B; \ \mu B \ast= \text{au};\n";
cout << "p: \text{[atm]}:\_;\ cin >> p;\n";
cout << "T: \text{[K]}:\_;\ cin >> t;\n";
cout << "V: \text{[ml/mol]}:\_;\ cin >> v; \ v \ast= (1.0e21/\text{NA}_c)\ast n;\n";
cout << "disp: \text{[nm]}:\_;\ cin >> disp;\n";
cout << "dv/v: \_;\ cin >> dvv;\n";
cout << "pvc: \_;\ cin >> pvch;\n";
cout << "pswp: \_;\ cin >> pswp;\n";
cout << "dr: \text{[nm]}:\_;\ cin >> dr;\n";

nt = 0;
cout << "ntskip: \_;\ cin >> ntskip;\n";
cout << "ntprint/ntskip: \_;\ cin >> ntprint;\n";
cout << "ntjob/ntskip: \_;\ cin >> ntjob;\n";

// allocate arrays

float c = cb(\nu);
for (int i = 0; i < n; ++i) { // np beads per particle
  x[i] = new float[\nu];
  y[i] = new float[\nu];
  z[i] = new float[\nu];
}

// random positions/coincident polymer beads

double step = c / n; // length between beads
for (int i = 0; i < n; ++i) { // np beads per particle
  float x = x[i];
  float y = y[i];
  float z = z[i];
  float r = the_rng(); // random number
  float r = cbrt(v); // cube root of volume
  for (int k = 0; k < np; ++k) {
    x[i][k] = x + r;
    y[i][k] = y + r;
    z[i][k] = z + r;
  }
}
// write startup file

ofstream fout("in.dat");

fout << n << 'n'
    << nA << 'n'
    << np << 'n'
    << nt << 'n'
    << ntjob << 'n'
    << ntprint << 'n'
    << ntskip << 'n'
    << the_rng << 'n'
    << disp << 'n'
    << dr << 'n'
    << epsA << 'n'
    << sigA << 'n'
    << mA << 'n'
    << epsB << 'n'
    << sigB << 'n'
    << mB << 'n'
    << p << 'n'
    << t << 'n'
    << v << 'n'
    << dvv << 'n'
    << pvch << 'n'
    << pswp << 'n';

    for (int i = 0; i < n; ++i)
        for (int k = 0; k < np; ++k)
            fout << x[i][k] << 'n';

    for (int i = 0; i < n; ++i)
        for (int k = 0; k < np; ++k)
            fout << y[i][k] << 'n';

    for (int i = 0; i < n; ++i)
        for (int k = 0; k < np; ++k)
            fout << z[i][k] << 'n';

fout.close();

return 0;
} // main

C.1.2 Lattice Initial Configuration: lnptpimclj

lnptpimclj creates an fcc lattice as initial configuration for NpT Path-Integral Monte Carlo of Lennard-Jones mixtures. It is checked that the number of particles n is a magic number for an fcc lattice.
C.1 Initialization and Re-Initialization

// lnpptimelj.cpp
// create initial configuration (fcc lattice) for NpT Path-Integral
// Monte Carlo of Lennard-Jones mixtures
//
// the following units are used:
// Length: 1 nm
// Volume: 1 nm**3
// Density: 1/nm**3
// Energy: 1 kJ/mol
// Pressure: 1 atm
// Temperature: 1 K
//
#include <math.h>
#include <stdlib.h>
#include <iostream.h>
#include <fstream.h>
#include "float_rng.h"

// global data; to be stored in the .dat files
int n; // total number of particles
int nA; // number of particles of substance A
int np; // Trotter number (beads/particle)
float epsA; // LJ-epsilon (in kJ/mol) for substance A
float sigA; // LJ-sigma (in nm) for substance A
float mA; // particle mass (in kg) for substance A
float epsB; // LJ-epsilon (in kJ/mol) for substance B
float sigB; // LJ-sigma (in nm) for substance B
float mB; // particle mass (in kg) for substance B
float p; // pressure
float t; // temperature
float v; // volume
float disp; // particle displacement
float dvv; // volume change per volume dv/v
float pvch; // probability for volume change
float pswp; // probability for swapping of particles
float dr; // channel width for pair-correlation
long nt; // "time"
```c
int ntskip; // thinning factor
int ntprint; // print control variables every ntprint*ntskip passes
int ntjob; // ntjob*ntskip passes

float ** x; // x-coordinates for the n particles
float ** y; // y-coordinates for the n particles
float ** z; // z-coordinates for the n particles

float _rng the_rng; // the RNG

// global data; not to be stored in the .dat files
const float au_c = 1.66053873e-27; // atomic mass unit in kg
const float NA_c = 6.02214199e23; // Avogadro's constant in 1/mol
const float R_c = 8.314472e-3; // molar gas constant in kJ/(mol K)

// main
int main()
{
    cout << "n: "; cin >> n;

    int nm = 0;
    for ( ; ; ) {
        ++nm;
        if (nm*nm*nm == 2*n) break;
        else if (nm*nm*nm > 2*n) {
            cout << "n is not a magic number" << endl;
            exit(1);
        }
    }

    cout << "nA: "; cin >> nA; cout << "chiA== " << float(nA)/n << endl;
    cout << "np: "; cin >> np;

    if (((np%2 != 0) && (np != 1)) { 
        cout << "np has to be even or == 1" << endl;
        exit(1);
    }

    cout << "A: eps/k[K]: "; cin >> epsA; epsA *= R_c;
    cout << "A: sig[nm]: "; cin >> sigA;
    cout << "A: m[au]: "; cin >> mA; mA *= au_c;
    cout << "B: eps/k[K]: "; cin >> epsB; epsB *= R_c;
    cout << "B: sig[nm]: "; cin >> sigB;
    cout << "B: m[au]: "; cin >> mB; mB *= au_c;

    cout << "p[atm]: "; cin >> p;
    cout << "T[K]: "; cin >> t;
```
C.1 Initialization and Re-Initialization

cout << "V [ml/mol]: "; cin >> v; v *= (1.0e21/NA_c)*n;
cout << "disp [nm]: "; cin >> disp;
cout << "dv/v: "; cin >> dvv;
cout << "pvch: "; cin >> pvch;
cout << "pswp: "; cin >> pswp;
cout << "dr [nm]: "; cin >> dr;

nt = 0;
cout << "ntskip: "; cin >> ntskip;
cout << "ntprint/ntskip: "; cin >> ntprint;
cout << "ntjob/ntskip: "; cin >> ntjob;

// allocate arrays

x = new (float*)[n];
y = new (float*)[n];
z = new (float*)[n];
for (int i = 0; i < n; ++i) { // np beads per particle
    x[i] = new float[np];
    y[i] = new float[np];
    z[i] = new float[np];
}

// fcc lattice/coincident polymer beads

float c = cbrt(v);
int j = 0;
for (int jx = 0; jx <= mm-1; ++jx)
    for (int jy = 0; jy <= mm-1; ++jy)
        for (int jz = 0; jz <= mm-1; ++jz)
            if (((jx+jy+jz)%2 == 0) {
                for (int k = 0; k < np; ++k) {
                    x[j][k] = (((jx+0.5)/mm-0.5)*c;
                    y[j][k] = (((jy+0.5)/mm-0.5)*c;
                    z[j][k] = (((jz+0.5)/mm-0.5)*c;
                }
                ++j;
            }

// write startup file

ofstream fout("in.dat");

fout << n << 
    << nA << 

115
C The Code

<< np << 'w'
<< nt << 'w'
<< ntjob << 'w'
<< ntprint << 'w'
<< ntskip << 'w'
<< the_rng << 'w'
<< disp << 'w'
<< dr << 'w'
<< epsA << 'w'
<< sigA << 'w'
<< mA << 'w'
<< epsB << 'w'
<< sigB << 'w'
<< mB << 'w'
<< p << 'w'
<< t << 'w'
<< v << 'w'
<< dvv << 'w'
<< pvch << 'w'
<< pswp << 'w';

for (int i = 0; i < n; ++i)
  for (int k = 0; k < np; ++k)
    fout << x[i][k] << 'w';
for (int i = 0; i < n; ++i)
  for (int k = 0; k < np; ++k)
    fout << y[i][k] << 'w';
for (int i = 0; i < n; ++i)
  for (int k = 0; k < np; ++k)
    fout << z[i][k] << 'w';

fout.close();

return 0;
} // main

C.1.3 Re-Initialization: znptpimclj

znptpimclj re-initializes the startup/checkpoint file for \textit{NpT Path-Integral Monte Carlo of Lennard-Jones mixtures}.

// // znptpimclj.cpp
//--
//-- \textit{NpT Path-Integral Monte Carlo of Lennard–Jones mixtures}
//-- re-initialize startup/checkpoint file
//--
//-- the following units are used:
//-- Length: 1 nm

116
C.1 Initialization and Re-Initialization

// Volume: 1 nm**3
// Density: 1/nm**3
// Energy: 1 kJ/mol
// Pressure: 1 atm
// Temperature: 1 K

#include <math.h>
#include <string.h>
#include <iostream.h>
#include <fstream.h>
#include "float_rng.h"

// global data; to be stored in the .dat files
int n; // total number of particles
int nA; // number of particles of substance A
int np; // Trotter number (beads/particle)

float epsA; // LJ-epsilon (in kJ/mol) for substance A
float sigA; // LJ-sigma (in nm) for substance A
float mA; // particle mass (in kg) for substance A
float epsB; // LJ-epsilon (in kJ/mol) for substance B
float sigB; // LJ-sigma (in nm) for substance B
float mB; // particle mass (in kg) for substance B

float p; // pressure
float t; // temperature
float v; // volume

float disp; // particle displacement

float dvv; // volume change per volume dv/v
float pvch; // probability for volume change

float pswp; // probability for swapping of particles

float dr; // channel width for pair-correlation

long nt; // "time"
int ntskip; // thinning factor
int ntprint; // print control variables every ntprint*ntskip passes
int ntjob; // ntjob*ntskip passes

float** x; // x-coordinates for the n particles
float** y; // y-coordinates for the n particles
float** z; // z-coordinates for the n particles
float_rng the_rng; // the RNG

// global data; not to be stored in the .dat files

const float au_c = 1.66053873e−27; // atomic mass unit in kg
const float NA_c = 6.02214199e23; // Avogadro’s constant in 1/mol
const float R_c = 8.314472e−3; // molar gas constant in kJ/(mol K)

// global functions (declarations)

float center_of_mass(const float * v);

template <class t> void getval(t& val);

// main

int main()
{
    char fname[100];
    cout << "infile [out.dat]: " ;
    cin.getline(fname, 100, '\n');
    if (strcmp(fname,"") == 0)
        strcpy(fname, "out.dat");

    ifstream fin(fname);

    fin >> n >> nA >> np >> nt
        >> ntjob >> nprint >> ntskip
        >> the_rng
        >> disp >> dr
        >> epsA >> sigA >> mA
        >> epsB >> sigB >> mB
        >> p >> t >> v
        >> dvv >> pvch >> pswp;

    // allocate arrays

    x = new (float*)[n];
    y = new (float*)[n];
    z = new (float*)[n];
    for (int i = 0; i < n; ++i) { // np beads per particle
        x[i] = new float[np];
        y[i] = new float[np];
        z[i] = new float[np];
    }

    // positions

    for (int i = 0; i < n; ++i)
C.1 Initialization and Re-Initialization

```c
for (int k = 0; k < np; ++k)
    fin >> x[i][k];
for (int i = 0; i < n; ++i)
    for (int k = 0; k < np; ++k)
        fin >> y[i][k];
for (int i = 0; i < n; ++i)
    for (int k = 0; k < np; ++k)
        fin >> z[i][k];
fin.close();

// input new values

float vn = v;

cout << "n==" << n << endl;
cout << "nA==" << nA << endl;
cout << "np==" << np << endl;
epsA /= Rc; cout << "A:eps/k==" << epsA << "J/K:"
    getval(epsA); epsA *= Rc;
cout << "A:sign==" << sigA << "J/mol:"
    getval(sigA);
mA /= au_c; cout << "A:m==" << mA << "au:
    getval(mA); mA *= au_c;
epsB /= Rc; cout << "B:eps/k==" << epsB << "J/K:"
    getval(epsB); epsB *= Rc;
cout << "B:sign==" << sigB << "J/mol:"
    getval(sigB);
mB /= au_c; cout << "B:m==" << mB << "au:
    getval(mB); mB *= au_c;

cout << "p==" << p << "atm:"
    getval(p);
cout << "t==" << t << "K:"
    getval(t);
vn /= (1.0e21/NA_c)*n; cout << "v==" << vn << "ml/mol:"
    getval(vn); vn *= (1.0e21/NA_c)*n;
cout << "disp==" << disp << "nm:"
    getval(disp);
cout << "dv/v==" << dvv << ""
    getval(dvv);
cout << "pvch==" << pvch << ""
    getval(pvch);
cout << "pswp==" << pswp << ""
    getval(pswp);
```

getval(pswp);

cout << "dr[==]" << dr << "nm]":ω; getval(dr);

cout << "nts[==]" << nts << " ]":ω; getval(nts);

cout << "ntpr/nts[==]" << ntpr << " ]":ω; getval(ntpr);

cout << "ntjo/nts[==]" << ntjo << " ]":ω; getval(ntjo);

// rescale positions

if (vn != v) {
    float fact = cbnc(vn/v);
    v = vn;
    for (int i = 0; i < n; ++i) {
        float xcm = centroid_mass(x[i]);
        float ycm = centroid_mass(y[i]);
        float zcm = centroid_mass(z[i]);
        for (int k = 0; k < np; ++k) {
            x[i][k] = (x[i][k]−xcm)+fact*xcm;
            y[i][k] = (y[i][k]−ycm)+fact*ycm;
            z[i][k] = (z[i][k]−zcm)+fact*zcm;
        }
    }
}

// write new startup file

ofstream fout("in.dat");

nt = 0;

cout << n << ' ';

<< nA << ' ';

<< np << ' ';

<< nt << ' ';

<< ntjo << ' ';

<< ntpr << ' ';

<< nts << ' ';

<< the_rng << ' ';

<< disp << ' ';

<< dr << ' ';

<< epsA << ' ';

<< sigA << ' ';

<< mA << ' ';

<< epsB << ' ';

<< sigB << ' ';

120
C.2 Simulation

```
<< mB << 'ω'
<< p << 'ω'
<< t << 'ω'
<< v << 'ω'
<< dv << 'ω'
<< pvch << 'ω'
<< pswp << 'ω';
```

```
for (int i = 0; i < n; ++i)
    for (int k = 0; k < np; ++k)
        fout << x[i][k] << 'ω';
for (int i = 0; i < n; ++i)
    for (int k = 0; k < np; ++k)
        fout << y[i][k] << 'ω';
for (int i = 0; i < n; ++i)
    for (int k = 0; k < np; ++k)
        fout << z[i][k] << 'ω';
```

```
fout.close();
```

```
return 0;
} // main
```

```
float center_of_mass(const float* v)
{
    float sum = 0.0;
    for (int k = 0; k < np; ++k)
        sum += v[k];
    return sum/np;
} // center_of_mass
```

```
template <class t> void getval(t& val)
{
    char c;
    cin.get(c);
    if (c != 'n') {
        cin.putback(c);
        cin >> val;
        cin.get(c);
    }
} // getval
```

C.2 Simulation

nptpimclj is the simulation program itself.

//
// nptpimclj.cpp
//
The Code

// NpT Path–Integral Monte Carlo of Lennard–Jones mixtures
// primitive algorithm
// direct sampling of polymer beads (FT-method)
// crude energy estimator
//
// the following units are used:
// Length: 1 nm
// Volume: 1 nm**3
// Density: 1/nm**3
// Energy: 1 kJ/mol
// Pressure: 1 atm
// Temperature: 1 K
//
#include <float.h>
#include <math.h>
#include <stdlib.h>
#include <iostream.h>
#include <fstream.h>
#include "float_rng.h"

// global data; to be stored in the .dat files
int n; // total number of particles
int nA; // number of particles of substance A
int np; // Trotter number (beads/particle)
float epsA; // LJ–epsilon (in kJ/mol) for substance A
float sigA; // LJ–sigma (in nm) for substance A
float mA; // particle mass (in kg) for substance A
float epsB; // LJ–epsilon (in kJ/mol) for substance B
float sigB; // LJ–sigma (in nm) for substance B
float mB; // particle mass (in kg) for substance B

float p; // pressure
float t; // temperature
float v; // volume

float disp; // particle displacement

float dvv; // volume change per volume dv/v
float pvch; // probability for volume change

float pswp; // probability for swapping of particles

float dr; // channel width for pair-correlation
int ndr; // number of channels in the histogramm
C.2 Simulation

long nt; // "time"
int ntskip; // thinning factor
int ntprint; // print control variables every ntprint*ntskip passes
int ntjob; // ntjob*ntskip passes

float** x; // x-coordinates for the n particles
float** y; // y-coordinates for the n particles
float** z; // z-coordinates for the n particles

float_rng the_rng; // the RNG

double accrpA; // accumulated acceptance ratio for particle moves for subst. A
double accrpB; // accumulated acceptance ratio for particle moves for subst. B
double accrv; // accumulated acceptance ratio for volume moves
double accrs; // accumulated acceptance ratio for particle swaps
double arho; // accumulated density

double ak; // accumulated kinetic energy for mixture

double muAA; // accumulated potential energy for interaction A<>A
double muAB; // accumulated potential energy for interaction A<>B

double bh; // accumulated enthalpy for mixture

double ar; // accumulated volume

double ar2A; // accumulated r^2 for substance A
double ar2B; // accumulated r^2 for substance B

double* agAA; // accumulated pair-correlation function for distance A<>A

double* agAB; // accumulated pair-correlation function for distance A<>B

double* agBB; // accumulated pair-correlation function for distance B<>B

// global data; not to be stored in the .dat files

const float NA_c = 6.02214199e23; // Avogadro's constant in 1/mol
const float kB_c = 1.3806503e23; // Boltzmann's constant in J/K
const float R_c = 8.314472e-3; // molar gas constant in kJ/(mol K)
const float h_c = 6.62606876e-34; // Planck's constant in Js

const float alnmax = log(0.1*FLT_MAX);
const float pi = 4.0*atan(1.0);

int nB; // number of particles of substance B

float epsAB; // LJ-epsilon (in kJ/mol) for interaction A<>B
float sigAB; // LJ-sigma (in nm) for interaction A<>B

float*** uatt; // interaction matrix; attractive part of LJ potential
float*** urep; // repulsive part of LJ potential

float u12AA; // factor for repulsive part of LJ potential for interaction A<>A
float u6AA; // factor for attractive part of LJ potential for interaction A<>A
C The Code

float u12AB; // factor for repulsive part of LJ potential for interaction A<->B
float u6AB; // factor for attractive part of LJ potential for interaction A<->B
float u12BB; // factor for repulsive part of LJ potential for interaction B<->B
float u6BB; // factor for attractive part of LJ potential for interaction B<->B

float beta; // 1/kT in reduced units
float qA; // \(\hbar^2/2m\) in reduced units for substance A
float qB; // \(\hbar^2/2m\) in reduced units for substance B

float c; // length of simulation cube
float c2; // == c/2
float c2m; // == -c/2
float r2max; //

float rc2; // square of cutoff radius
float su0AA; // tail correction for potential energy for substance A
float su0AB; // tail correction for potential energy for interaction A<->B
float su0BB; // tail correction for potential energy for substance B

long ntrypA; // tried particle moves for substance A
long ntrypB; // tried particle moves for substance B
long ntryv; // tried volume moves
long ntrys; // tried particle swaps
long naccpA; // accepted particle moves for substance A
long naccpB; // accepted particle moves for substance B
long naccv; // accepted volume moves
long naccs; // accepted particle swaps

// auxiliary arrays to avoid repeated allocation and deallocation

float ** uattn;
float ** urepn;
float ** uattnA;
float ** urepnA;
float ** uattnB;
float ** urepnB;

// DFT-arrays for the generation of the beads

float * sqvarA; // sqrt of the variances for substance A
float * sqvarB; // sqrt of the variances for substance B
float ** xmat; // transformation matrix

// inline functions

template<class T> inline T max(const T a, const T b)
{
    return (a >= b) ? a : b;
} // max(T, T)
template<class T> inline T min(const T a, const T b)
{
    return (a <= b) ? a : b;
} // min(T, T)

template<class T> inline T sqr(const T x)
{
    return x*x;
} // sqr(T)

template<class T> inline T cube(const T x)
{
    return x*x*x;
} // cube(T)

// global functions (declarations)

// functions called from main

void get_cf();
void u_init();
void par_move();
void vol_move();
void par_swap();
void means();
void put_cf();

// auxiliary functions

void pbc(float& dx, float& dy, float& dz);
float center_of_mass(const float *const v);
void new_beads(float *const v, const float *const sqvar);

// main

int main()
{
    // read startup/checkpoint file & initialize interaction matrix

get_cf();
u_init();

    // do (ntskip*ntjob) passes/volume changes...

for (int i = 0; i < ntjob; ++i) {
    ++nt;
    for (int j = 0; j < ntskip*n; ++j) {
        float ran = the_rng();
    }
}
if (ran < pvch) vol_move();
else if (ran < pvch+pswp) par_swap();
else par_move();
}
means();

// write checkpoint file
put_cf();
return 0;
} // main

// global functions (definitions)
// functions called from main

void get_cf()
// read startup/checkpoint file
{
    ifstream fin("in.dat");
    fin >> n >> nA >> np >> nt
    >> njob >> nprint >> ntskip
    >> the_rng
    >> disp >> dr
    >> epsA >> sigA >> mA
    >> epsB >> sigB >> mB
    >> p >> t >> v
    >> dmv >> pvch >> pswp;

    nB = n-nA;

    // Lorentz−Berthelot
    epsAB = sqrt(epsA*epsB);
    sigAB = 0.5*(sigA+sigB);

    // constant factors for potential energy
    u12AA = 4.0*epsA*cube(sqr(sqr(sigA)));
    u6AA = 4.0*epsA*cube(sqr(sigA));
    u12AB = 4.0*epsAB*cube(sqr(sqr(sigAB)));
    u6AB = 4.0*epsAB*cube(sqr(sigAB));
    u12BB = 4.0*epsB*cube(sqr(sqr(sigB)));
    u6BB = 4.0*epsB*cube(sqr(sigB));
    beta = 1.0/(R_c*t); // in 1/(kJ/mol)
\[ q_A = 1.0 \times 10^{15} N_A c \times \sqrt{\frac{h_c}{2.0 \times \pi}} / (2.0 \times m_A); \quad \text{in kJ/mol nm}^{2} \]
\[ q_B = 1.0 \times 10^{15} N_A c \times \sqrt{\frac{h_c}{2.0 \times \pi}} / (2.0 \times m_B); \quad \text{in kJ/mol nm}^{2} \]

// pair-correlation

\[ \text{ndr} = \text{int}(0.5 \times \text{cbrt(v)}/\text{dr}); \quad \text{g(r) up to half box length} \]
\[ \text{agAA} = \text{new double}[\text{ndr}]; \]
\[ \text{agAB} = \text{new double}[\text{ndr}]; \]
\[ \text{agBB} = \text{new double}[\text{ndr}]; \]

// uatt and urep are (n x n x np)

\[ \text{uatt} = \text{new float*}[\text{n}]; \]
\[ \text{urep} = \text{new float*}[\text{n}]; \]
\[ \text{for (int i = 0; i < n; ++i) \{} \]
\[ \quad \text{uatt}[i] = \text{new float*}[\text{n}]; \]
\[ \quad \text{urep}[i] = \text{new float*}[\text{n}]; \]
\[ \quad \text{for (int j = 0; j < n; ++j) \{} \]
\[ \quad \quad \text{uatt}[i][j] = \text{new float}[\text{np}]; \]
\[ \quad \quad \text{urep}[i][j] = \text{new float}[\text{np}]; \]
\[ \text{\}} \]

// x, y and z are (n x np)

\[ \text{x} = \text{new float*}[\text{n}]; \]
\[ \text{y} = \text{new float*}[\text{n}]; \]
\[ \text{z} = \text{new float*}[\text{n}]; \]
\[ \text{for (int i = 0; i < n; ++i) \{} \quad \text{np beads per particle} \]
\[ \quad \text{x}[i] = \text{new float}[\text{np}]; \]
\[ \quad \text{y}[i] = \text{new float}[\text{np}]; \]
\[ \quad \text{z}[i] = \text{new float}[\text{np}]; \]
\[ \text{\}} \]

// positions & accumulated averages

\[ \text{for (int i = 0; i < n; ++i) \{} \]
\[ \quad \text{for (int k = 0; k < np; ++k) \{} \]
\[ \quad \quad \text{fin >> x[i][k];} \]
\[ \quad \quad \text{for (int k = 0; k < np; ++k) \{} \]
\[ \quad \quad \quad \text{fin >> y[i][k];} \]
\[ \quad \quad \text{for (int k = 0; k < np; ++k) \{} \]
\[ \quad \quad \quad \text{fin >> z[i][k];} \]
\[ \quad \text{if (nt > 0) \{} \]
\[ \quad \quad \text{fin >> accrA >> accrB >> accrV >> accrs >> arho} \]
\[ \quad \text{\}} \]

127
C The Code

```c
>> ak
>> auAA >> auAB >> auBB
>> ah
>> av
>> ar2A >> ar2B;
fin >> nmr;
for (int i = 0; i < m; ++i)
  fin >> agAA[i] >> agAB[i] >> agBB[i];
}
else {
  accpA = accpB = accv = accr = arho = ak = auAA = auAB = auBB = ah = av = ar2A = ar2B = 0.0;
  for (int i = 0; i < m; ++i)
    agAA[i] = agAB[i] = agBB[i] = 0.0;
}
fin.close();

// auxiliary arrays
uattn = new (float*)[n];
urepn = new (float*)[n];
uattNA = new (float*)[n];
urepNA = new (float*)[n];
uattNB = new (float*)[n];
urepNB = new (float*)[n];
for (int i = 0; i < n; ++i) {
  uattn[i] = new float[np];
  urepn[i] = new float[np];
  uattNA[i] = new float[np];
  urepNA[i] = new float[np];
  uattNB[i] = new float[np];
  urepNB[i] = new float[np];
}

// DFT arrays
sqvarA = new float[np];
sqvarB = new float[np];
xmat = new (float*)[np];
for (int k = 0; k < np; ++k)
  xmat[k] = new float[np];

// DFT
```

128
for (int i = 1; i < np-1; i += 2) {
    int k = (i+1)/2;
    sqvarA[i] = sqvarA[i+1] = sqrt(beta*qA/(2.0*sqr(np)*(1.0-cos(2.0*pi*k np))));
    sqvarB[i] = sqvarB[i+1] = sqrt(beta*qB/(2.0*sqr(np)*(1.0-cos(2.0*pi*k np))));
}

for (int i = 0; i < np; ++i)
    for (int j = 1; j < np-1; j += 2) {
        int k = (j+1)/2;
        float arg = (2.0*pi*i)/np*k;
        xmat[i][j] = 2.0*cos(arg);
        xmat[i][j+1] = -2.0*sin(arg);
    }

// box parameters

c = cbrt(v);
c2 = 0.5*c;
c2m = -c2;
r2max = sqr(c2);

// check positions

for (int i = 0; i < n; ++i)
    if (max(fabs(center_of_mass(x[i])),
            max(fabs(center_of_mass(y[i])),
                fabs(center_of_mass(z[i])))) > 1.001*c2)
        exit(1);

// cutoff & tail corrections

float rc = c2;
rc2 = sqr(rc);

// the total tail correction is su0 = su0AA+su0AB+su0BB, which reduces to
// su0 = 2.0*pi*sqr(n)/v*np*(u12/(9.0*cube(cube(rc)))-u6/(3.0*cube(cube(rc))));
// in the pure case
// the additional factor 2.0 in su0AB comes from
// sqr(n) = sqr(nA+nB) = sqr(nA)+2.0*nA*nB+sqr(nB)
su0AA = 2.0*pi*sqr(nA)/v*np* (u12AA/(9.0*cube(cube(rc)))-u6AA/(3.0*cube(cube(rc))));
su0AB = 4.0*pi*nA*nB/v*np
    (u12AB/(9.0*cube(cube(rc)))-u6AB/(3.0*cube(rc)));

su0BB = 2.0*pi*sqr(nB)/v*np
    (u12BB/(9.0*cube(cube(rc)))-u6BB/(3.0*cube(rc)));

// clear acceptance counter

ntrypA = ntrypB = ntrvy = ntrys = 0;
naccpA = naccpB = naccv = naccs = 0;
}

// get cf

void u_init()

    // initialize interaction matrix
    {
        for (int k = 0; k < np; ++k) { // beads

            for (int i = 0; i < nA-1; ++i) { // pairs A<>A
                urep[i][i][k] = uatt[i][i][k] = 0.0;
                for (int j = i+1; j < nA; ++j)
                    {
                        float dx = x[j][k]-x[i][k];
                        float dy = y[j][k]-y[i][k];
                        float dz = z[j][k]-z[i][k];
                        pbc(dx, dy, dz);
                        float r2 = sqr(dx)+sqr(dy)+sqr(dz);
                        if (r2 < rc2) {
                            float rm6 = 1.0/cube(r2);
                            urep[i][j][k] = u12AA*sqr(rm6);
                            uatt[i][j][k] = -u6AA*rm6;
                        }
                        else urep[i][j][k] = uatt[i][j][k] = 0.0;
                        uatt[j][i][k] = uatt[i][j][k];
                        urep[j][i][k] = urep[i][j][k];
                    }
        }

        urep[nA-1][nA-1][k] = uatt[nA-1][nA-1][k] = 0.0;

        for (int i = nA; i < n; ++i) { // pairs A<>B and B<>A
            for (int j = 0; j < nA; ++j)
                {
                    float dx = x[j][k]-x[i][k];
                    float dy = y[j][k]-y[i][k];
                    float dz = z[j][k]-z[i][k];
                    pbc(dx, dy, dz);
                    float r2 = sqr(dx)+sqr(dy)+sqr(dz);
                    if (r2 < rc2) {
                        float rm6 = 1.0/cube(r2);
                        urep[i][j][k] = u12AB*sqr(rm6);
                        uatt[i][j][k] = -u6AB*rm6;
                    }
                }
    }
C.2 Simulation

```c
else urep[i][j][k] = uatt[i][j][k] = 0.0;
    uatt[j][i][k] = uatt[i][j][k];
    urep[j][i][k] = urep[i][j][k];
} // pairs A<>B and B<>A

for (int i = nA; i < n-1; ++i) { // pairs B<>B
    urep[i][i][k] = uatt[i][i][k] = 0.0;
    for (int j = i+1; j < n; ++j) {
        float dx = x[j][k]-x[i][k];
        float dy = y[j][k]-y[i][k];
        float dz = z[j][k]-z[i][k];
        pbc(dx, dy, dz);
        float r2 = sqr(dx)+sqr(dy)+sqr(dz);
        if (r2 < rc2) {
            float rm6 = 1.0/cube(r2);
            urep[i][j][k] = u12BB*sqr(rm6);
            uatt[i][j][k] = -u6BB*rm6;
        }
        else urep[i][j][k] = uatt[i][j][k] = 0.0;
        uatt[j][i][k] = uatt[i][j][k];
        urep[j][i][k] = urep[i][j][k];
    }
} // pairs B<>B
    urep[n-1][n-1][k] = uatt[n-1][n-1][k] = 0.0;

} // beads
} // u_init

void par_move()
// particle move
{
// random particle

    int i = min(int(the_rng()*n), n-1); // 0 <= i < n

// move particle of substance A or B?

    float* sqvar; // the variances for the new beads
    float u12A; // factors for pairs i<>A
    float u6A;
    float u12B; // factors for pairs i<>B
    float u6B;
    if (i < nA) { // move particle of substance A
        ++ntrypA;
        sqvar = sqvarA;
    }
    u12A = u12AA;
```
C The Code

```c
u6A = u6AA;
u12B = u12AB;
u6B = u6AB;
} else { // move particle of substance B
  ++ntrypB;
  sqvar = sqvarB;
  u12A = u12AB;
  u6A = u6AB;
  u12B = u12BB;
  u6B = u6BB;
}

// displace center of mass

float xcmn = center_of_mass(x[i])+(the_rng()-0.5)*disp;
float ycmn = center_of_mass(y[i])+(the_rng()-0.5)*disp;
float zcmn = center_of_mass(z[i])+(the_rng()-0.5)*disp;
pbc(xcmn, ycmn, zcmn);

static float* xn = new float[np];
static float* yn = new float[np];
static float* zn = new float[np];
new_beads(xn, sqvar);
new_beads(yn, sqvar);
new_beads(zn, sqvar);

// new polymer coordinates

for (int k = 0; k < np; ++k) {
  xn[k] += xcmn;
  yn[k] += ycmn;
  zn[k] += zcmn;
}

// intermolecular (potential) energy difference

float su = 0.0;
for (int k = 0; k < np; ++k) { // beads
  for (int j = 0; j < nA; ++j) { // pairs i<>A
    if (j != i) {
      float dx = x[j][k]-xn[k];
      float dy = y[j][k]-yn[k];
      float dz = z[j][k]-zn[k];
pbc(dx, dy, dz);

      float r2 = sqr(dx)+sqr(dy)+sqr(dz);
      if (r2 < rc2) {
        float rm6 = 1.0/cube(r2);
```
C.2 Simulation

\[ \text{urepn}[j][k] = u_{12A} \times \text{sqr}(r_{m6}); \]
\[ \text{uattn}[j][k] = -u_{6A} \times r_{m6}; \]

\} else \text{ urepn}[j][k] = \text{uattn}[j][k] = 0.0; \]
\[ \text{su} += (\text{urepn}[j][k]+\text{uattn}[j][k]) - (\text{urep}[i][j][k]+\text{uatt}[i][j][k]); \]

\} else \text{ urepn}[j][k] = \text{uattn}[j][k] = 0.0; \]

// pairs i<->A

\for (\text{int} j = nA; j < n; ++j) { // pairs i<->B
\if (j != i) { \text{float} dx = x[j][k] - xn[k]; \text{float} dy = y[j][k] - yn[k]; \text{float} dz = z[j][k] - zn[k]; \text{pbc}(dx, dy, dz); \text{float} r2 = \text{sqr}(dx) + \text{sqr}(dy) + \text{sqr}(dz); \text{if} (r2 < r_{c2}) { \text{float} rm6 = 1.0 / \text{cube}(r2); \text{urepn}[j][k] = u_{12B} \times \text{sqr}(rm6); \text{uattn}[j][k] = -u_{6B} \times rm6; \} else \text{ urepn}[j][k] = \text{uattn}[j][k] = 0.0; \text{su} += (\text{urepn}[j][k]+\text{uattn}[j][k]) - (\text{urep}[i][j][k]+\text{uatt}[i][j][k]); \}

\} else \text{ urepn}[j][k] = \text{uattn}[j][k] = 0.0; \}

// beads
\text{su} /= \text{np};

// acceptance test
\text{bool} accepted;
\text{if} (\text{su} <= 0.0) \text{accepted} = \text{true};
\text{else} { \text{if} (\text{beta} \times \text{su} < \text{alnmax}) { \text{if} (\text{the_rng}() <= \text{exp}(-\text{beta} \times \text{su})) \text{accepted} = \text{true}; \text{else} \text{accepted} = \text{false}; \} \text{else} \text{accepted} = \text{false}; \}

// accept move & update interaction matrix
\text{if} (\text{accepted}) {
```c
if (i < nA) ++naccpA;
else ++naccpB;
for (int k = 0; k < np; ++k) {
    x[i][k] = xn[k];
    y[i][k] = yn[k];
    z[i][k] = zn[k];
}
for (int j = 0; j < n; ++j)
    for (int k = 0; k < np; ++k) {
        urep[i][j][k] = urep[j][i][k] = urepn[j][k];
        uatt[i][j][k] = uatt[j][i][k] = uattn[j][k];
    }
} // par_move

void vol_move()
    // volume move
{
    ++ntryv;

    float vn = v*(1.0+dv*(the_rng()-0.5));
    if (vn <= 0.0) return;

    float rcn = 0.5*cbrt(vn);
    float su0AA = 2.0*pi*sqr(nA)/vn*np*
        (u12AA/(9.0*cube(cube(rcn)))-u6AA/(3.0*cube(rcn)));
    float su0AB = 4.0*pi*nA*nB/vn*np*
        (u12AB/(9.0*cube(cube(rcn)))-u6AB/(3.0*cube(rcn)));
    float su0BB = 2.0*pi*sqr(nB)/vn*np*
        (u12BB/(9.0*cube(cube(rcn)))-u6BB/(3.0*cube(rcn)));

    // spring energy difference
    float susA = 0.0;
    for (int i = 0; i < nA; ++i) { // beads A
        int km1 = np-1;
        for (int k = 0; k < np; ++k) {
            susA += sqr(x[i][k]-x[i][km1])+
            sqr(y[i][k]-y[i][km1])+
            sqr(z[i][k]-z[i][km1]);
            km1 = k;
        }
    }
    susA *= (0.25*np/(sqr(beta)*qA));

    float susB = 0.0;
    for (int i = nA; i < n; ++i) { // beads B
        int km1 = np-1;
        for (int k = 0; k < np; ++k) {
```
C.2 Simulation

```c
susB += sqr(x[i][k]−x[i][km1])+ s2q(y[i][k]−y[i][km1])+ s2q(z[i][k]−z[i][km1]);
km1 = k;
}
susB *= (0.25*np/(sqr(beta)*qB));

float sus = (sqr(cbrt(vn/v))−1.0)*(susA+susB);

// scaling factors for potential energy
float frep = sqr(sqr(v/vn))−1.0;
float fatt = sqr(v/vn)−1.0;

// potential energy difference
float su = (su0AAn+su0ABn+su0BBn)−(su0AA+su0AB+su0BB);
for (int k = 0; k < np; ++k) // beads
    for (int i = 0; i < n−1; ++i) // pairs
        for (int j = i+1; j < n; ++j)
            su += frep*urep[i][j][k]+fatt*uatt[i][j][k];
su /= np;
su += sus+101325.0e−30*NA_c*p*(vn−v)−(n*np+1)/beta*log(vn/v);

// acceptance test
bool accepted;
if (su <= 0.0) accepted = true;
else {
    if (beta*su < alnmax) {
        if (the_rng() <= exp(−beta*su)) accepted = true;
        else accepted = false;
    }
    else accepted = false;
}

if (accepted) {
    ++naccv;

    // positions

    float fact = cbrt(vn/v);
    for (int i = 0; i < n; ++i)
        for (int k = 0; k < np; ++k) {
            x[i][k] *= fact;
            y[i][k] *= fact;
            z[i][k] *= fact;
        }
```

135
// box parameters

v = vn;
c = cbrt(v);
c2 = 0.5*c;
c2m = -c2;
r2max = sqr(c2);

ndr = min(ndr, int(c2/dr));

// cutoff & tail corrections

rc2 = sqr(rcn);
su0AA = su0AAn;
su0AB = su0ABn;
su0BB = su0BBn;

// update pair interaction matrix

u_init();
}
} // vol_move

void par_swap()
// particle swap
{
++ntrys;

// random particles

int iA = min(int(the_rng()*nA), nA−1); // 0 <= iA < nA
int iB = nA+min(int(the_rng()*nB), nB−1); // nA <= iB < nA+nB

// exchange only the center of mass

float xcmA = center_of_mass(x[iA]);
float ycmA = center_of_mass(y[iA]);
float xcmA = center_of_mass(z[iA]);
float xcmB = center_of_mass(x[iB]);
float ycmB = center_of_mass(y[iB]);
float xcmB = center_of_mass(z[iB]);

// new positions

static float* xnA = new float[np];
static float* ynA = new float[np];
static float* znA = new float[np];
static float * xnB = new float [np];
static float * ynB = new float [np];
static float * znB = new float [np];

for (int k = 0; k < np; ++k) {
    xnA[k] = xcmB+(x[iA][k]-xcmA);
    ynA[k] = ycmB+(y[iA][k]-ycmA);
    znA[k] = zcmB+(z[iA][k]-zcmA);
    xnB[k] = xcmA+(x[iB][k]-xcmB);
    ynB[k] = ycmA+(y[iB][k]-ycmB);
    znB[k] = zcmA+(z[iB][k]-zcmB);
}

float su = 0.0;
for (int k = 0; k < np; ++k) { // beads

    // energy change for particle iA

    for (int j = 0; j < nA; ++j) { // pairs iA<>A
        if (j != iA) {
            float dx = x[j][k]-xnA[k];
            float dy = y[j][k]-ynA[k];
            float dz = z[j][k]-znA[k];
            pbc(dx, dy, dz);

            float r2 = sqr(dx)+sqr(dy)+sqr(dz);
            if (r2 < rc2) {
                float rm6 = 1.0/cube(r2);
                urepnA[j][k] = u12AA*sqr(rm6);
                uattnA[j][k] = -u6AA*rm6;
            } else urepnA[j][k] = uattnA[j][k] = 0.0;
            su += (urepnA[j][k]+uattnA[j][k])-
                  (urep[iA][j][k]+uatt[iA][j][k]);
        }
    }

    // pairs iA<>B

    for (int j = nA; j < n; ++j) { // pairs iA<>B
        if (j != iB) {
            float dx = x[j][k]-xnA[k];
            float dy = y[j][k]-ynA[k];
            float dz = z[j][k]-znA[k];
            pbc(dx, dy, dz);

            float r2 = sqr(dx)+sqr(dy)+sqr(dz);
            if (r2 < rc2) {
                float rm6 = 1.0/cube(r2);
                urepnA[j][k] = u12AB*sqr(rm6);
            }
        }
    }
}

137
\begin{verbatim}
C The Code

\texttt{\begin{verbatim}
\texttt{uattnA[i][k] = \(- u6AB \times rm6;}
\} else \texttt{ urepnA[i][k] = uattnA[i][k] = 0.0;
\}
\texttt{su += (urepnA[i][k]+uattnA[i][k]) -
(urep[iA][i][k]+uatt[iA][i][k]);
\}
\texttt{else { \{
float dx = xnB[k]-xnA[k];
float dy = ynB[k]-ynA[k];
float dz = znB[k]-znA[k];
pbc(dx, dy, dz);
float r2 = sqr(dx)+sqr(dy)+sqr(dz);
if (r2 < rc2) {
float rm6 = 1.0/cube(r2);
urepnA[i][k] = u12AB\times squ(rm6);
uattnA[i][k] = \(- u6AB \times rm6;
\} else \texttt{ urepnA[i][k] = uattnA[i][k] = 0.0;
\}
\texttt{su += (urepnA[i][k]+uattnA[i][k]) -
(urep[iA][i][k]+uatt[iA][i][k]);
\}} // pairs iA<->B

// energy change for iB
\texttt{for (int j = 0; j < nA; ++j) { // pairs iB<->A
if (j != iA) { \{
float dx = x[j][k]-xnB[k];
float dy = y[j][k]-ynB[k];
float dz = z[j][k]-znB[k];
pbc(dx, dy, dz);
float r2 = sqr(dx)+sqr(dy)+sqr(dz);
if (r2 < rc2) {
float rm6 = 1.0/cube(r2);
urepnB[j][k] = u12AB\times squ(rm6);
uattnB[j][k] = \(- u6AB \times rm6;
\} else \texttt{ urepnB[j][k] = uattnB[j][k] = 0.0;
\}
\texttt{su += (urepnB[j][k]+uattnB[j][k]) -
(urep[iB][j][k]+uatt[iB][j][k]);
\}} else { \{
float dx = xnA[k]-xnB[k];
float dy = ynA[k]-ynB[k];
float dz = znA[k]-znB[k];
pbc(dx, dy, dz);
float r2 = sqr(dx)+sqr(dy)+sqr(dz);
if (r2 < rc2) {
\end{verbatim}}

\end{verbatim}
}
float rm6 = 1.0/cube(r2);
urepnB[j][k] = u12AB * sqr(rm6);
uattnB[j][k] = -u6AB * rm6;
}
else urepnB[j][k] = uattnB[j][k] = 0.0;
su += (urepnB[j][k] + uattnB[j][k]) -
    (urep[iB][j][k] + uatt[iB][j][k]);
}
// pairs iB<->A
for (int j = nA; j < n; ++j) { // pairs iB<->B
if (j != iB) {
    float dx = x[j][k] - xnB[k];
    float dy = y[j][k] - ynB[k];
    float dz = z[j][k] - znB[k];
    pbc(dx, dy, dz);
    float r2 = sqr(dx) + sqr(dy) + sqr(dz);
    float rm6 = 1.0/cube(r2);
    urepnB[j][k] = u12BB * sqr(rm6);
    uattnB[j][k] = -u6BB * rm6;
}
else urepnB[j][k] = uattnB[j][k] = 0.0;
su += (urepnB[j][k] + uattnB[j][k]) -
    (urep[iB][j][k] + uatt[iB][j][k]);
}
// pairs iA<->A
}
// beads
su /= np;

// acceptance test
bool accepted;
if (su <= 0.0) accepted = true;
else {
    if (beta * su < alnmax) {
        if (the_rng() <= exp(-beta * su)) accepted = true;
        else accepted = false;
    }
    else accepted = false;
}

// accept move & update interaction matrix
if (accepted) {
    ++naccs;
}
\begin{verbatim}
for (int k = 0; k < np; ++k) {
    x[iA][k] = xnA[k];
    y[iA][k] = ynA[k];
    z[iA][k] = znA[k];
    x[iB][k] = xnB[k];
    y[iB][k] = ynB[k];
    z[iB][k] = znB[k];
}
for (int j = 0; j < n; ++j)
    for (int k = 0; k < np; ++k) {
        urep[iA][j][k] = urep[j][iA][k] = urepA[j][k];
        uatt[iA][j][k] = uatt[j][iA][k] = uattA[j][k];
        urep[iB][j][k] = urep[j][iB][k] = urepB[j][k];
        uatt[iB][j][k] = uatt[j][iB][k] = uattB[j][k];
    }
}

// par_swap

void means()
    // calculate mean values
{
    // intramolecular (spring) energy
    float susA = 0.0;
    for (int i = 0; i < nA; ++i) { // beads A
        int kml = np-1;
        for (int k = 0; k < np; ++k) {
            susA += sqr(x[i][k]-x[i][kml]) +
                    sqr(y[i][k]-y[i][kml]) +
                    sqr(z[i][k]-z[i][kml]);
            kml = k;
        }
        susA *= (0.25*np/(sqr(beta)*qA));
    }
    susA = susA/n;

    float susB = 0.0;
    for (int i = nA; i < n; ++i) { // beads B
        int kml = np-1;
        for (int k = 0; k < np; ++k) {
            susB += sqr(x[i][k]-x[i][kml]) +
                    sqr(y[i][k]-y[i][kml]) +
                    sqr(z[i][k]-z[i][kml]);
            kml = k;
        }
        susB *= (0.25*np/(sqr(beta)*qB));
    }
    sus = (susA+susB)/n;
\end{verbatim}
C.2 Simulation

// radius of gyration

float sr2A = 0.0;
for (int i = 0; i < nA; ++i) {  // substance A
    float auxx = center_of_mass(x[i]);
    float auxy = center_of_mass(y[i]);
    float auxz = center_of_mass(z[i]);
    for (int k = 0; k < np; ++k)
        sr2A += sqr(x[i][k] - auxx) + sqr(y[i][k] - auxy) + sqr(z[i][k] - auxz);
}
sr2A /= nA*np;

float sr2B = 0.0;
for (int i = nA; i < n; ++i) {  // substance B
    float auxx = center_of_mass(x[i]);
    float auxy = center_of_mass(y[i]);
    float auxz = center_of_mass(z[i]);
    for (int k = 0; k < np; ++k)
        sr2B += sqr(x[i][k] - auxx) + sqr(y[i][k] - auxy) + sqr(z[i][k] - auxz);
}
sr2B /= nB*np;

// intermolecular (potential) energy & g(r)

float suAA = su0AA;
float suAB = su0AB;
float suBB = su0BB;
for (int k = 0; k < np; ++k) {  // beads
    for (int i = 0; i < nA-1; ++i) {  // pairs A<->A
        for (int j = i+1; j < nA; ++j)
            float dx = x[j][k] - x[i][k];
            float dy = y[j][k] - y[i][k];
            float dz = z[j][k] - z[i][k];
            pbc(dx, dy, dz);

            float r2 = sqr(dx) + sqr(dy) + sqr(dz);
            if (r2 < rc2)
                suAA += urep[i][j][k] + uatt[i][j][k];  // potential energy
            if (r2 < r2max)
                int ir = int(sqrt(r2)/dr);
                if (ir < ndr)
                    agAA[ir] += 1.0;  // g(r)
    }
}

for (int i = nA; i < n; ++i) {  // pairs A<->B and B<->A

}
for (int j = 0; j < nA; ++j) {
    float dx = x[j][k]−x[i][k];
    float dy = y[j][k]−y[i][k];
    float dz = z[j][k]−z[i][k];
pbc(dx, dy, dz);

    float r2 = sqr(dx)+sqr(dy)+sqr(dz);
    if (r2 < rc2)
        suAB += urep[i][j][k]+uatt[i][j][k]; // potential energy
    if (r2 < r2max) {
        int ir = int(sqrt(r2)/dr);
        if (ir < ndr)
            agAB[ir] += 1.0; // g(r)
    }
}

for (int i = nA; i < n-1; ++i) { // pairs A->B
    for (int j = i+1; j < n; ++j) {
        float dx = x[j][k]−x[i][k];
        float dy = y[j][k]−y[i][k];
        float dz = z[j][k]−z[i][k];
pbc(dx, dy, dz);

        float r2 = sqr(dx)+sqr(dy)+sqr(dz);
        if (r2 < rc2)
            suAB += urep[i][j][k]+uatt[i][j][k]; // potential energy
        if (r2 < r2max) {
            int ir = int(sqrt(r2)/dr);
            if (ir < ndr)
                agAB[ir] += 1.0; // g(r)
        }
    }
}

suAA /= n*np;
suAB /= n*np;
suBB /= n*np;

// kinetic energy

float sk = 1.5*np/beta−sus;

// print control variables

if ((ntprint > 0) && (nt%ntprint == 0))
    cout << nt << ' ' << double(naccpA)/max(double(ntrypA),1.0) << ' ' << ' ';

142
C.2 Simulation

```c
<< double(naccpB)/max(double(ntrypB),1.0) << ' '`
<< double(naccv)/max(double(ntrypv),1.0) << ' '`
<< double(naccs)/max(double(ntrys),1.0) << ' '`
<< sqrt(sr2A) << ' '`
<< sqrt(sr2B) << ' '`
<< n/v << ' '`
<< sk << ' '`
<< suAA+suAB+suBB << ' '`
<< (sk+suAA+suAB+suBB)+101325.0e-30*NA_c*p*v/n << ' '`
<< endl;

// accumulate averages

accrpA += double(naccpA)/max(double(ntrypA), 1.0);
accrpB += double(naccpB)/max(double(ntrypB), 1.0);
accrv += double(naccv)/max(double(ntrypv), 1.0);
accrs += double(naccs)/max(double(ntrys), 1.0);
arho += n/v;
ak += sk;
suAA += suAA;
suAB += suAB;
suBB += suBB;
ah += (sk+suAA+suAB+suBB)+101325.0e-30*NA_c*p*v/n;
av += v;
ar2A += sr2A;
ar2B += sr2B;

// reset acceptance counter

ntrypA = ntrypB = ntrypv = ntrys = 0;
naccpA = naccpB = naccv = naccs = 0;
}

// means

void put Cf() {
    // write checkpoint file
    ofstream fout("out.dat");
    fout << n << ' '
        << nA << ' '
        << np << ' '
        << nt << ' '
        << ntjob << ' '
        << ntprint << ' '
        << ntskip << ' '
        << the_rng << ' '
        << disp << ' '
        << dr << ' '
        << epsA << ' '
        << sigA << ' '
```
```c
<< mA << 'n'
<< epsB << 'n'
<< sigB << 'n'
<< mB << 'n'
<< t << 'n'
<< v << 'n'
<< dvv << 'n'
<< pvch << 'n'
<< pswp << 'n';

for (int i = 0; i < n; ++i)
    for (int k = 0; k < np; ++k)
        fout << x[i][k] << 'n';
for (int i = 0; i < n; ++i)
    for (int k = 0; k < np; ++k)
        fout << y[i][k] << 'n';
for (int i = 0; i < n; ++i)
    for (int k = 0; k < np; ++k)
        fout << z[i][k] << 'n';

fout << accrpA << 'n'
    << accrpB << 'n'
    << accrv << 'n'
    << accrs << 'n'
    << arho << 'n'
    << ak << 'n'
    << auAA << 'n'
    << auAB << 'n'
    << auBB << 'n'
    << ah << 'n'
    << av << 'n'
    << ar2A << 'n'
    << ar2B << 'n'
    << ndr << 'n';
for (int i = 0; i < ndr; ++i)
    fout << agAA[i] << 'n' << agAB[i] << 'n' << agBB[i] << 'n';

fout.close();
} // putcf

// auxiliary functions

void pbc(float& dx, float& dy, float& dz)
{
    if (dx > c2) dx -= c;
    else if (dx <= c2m) dx += c;
    if (dy > c2) dy -= c;
    else if (dy <= c2m) dy += c;
}```
if (dz > c2) dz -= c;
else if (dz <= c2m) dz += c;
} // pbc

float center_of_mass(const float *const v)
{
    float sum = 0.0;
    for (int k = 0; k < np; ++k)
        sum += v[k];
    return sum[np];
} // center_of_mass

float norm_random()
// return N(0,1)-distributed random numbers (Box-Muller method)
{
    static bool new_random = true;
    static float r;
    static float phi;

    if (new_random) {
        new_random = false;
        r = sqrt(-2.0*log(1.0-the_rng()));
        phi = 2.0*pi*the_rng();
        return r*cos(phi);
    }
    else {
        new_random = true;
        return r*sin(phi);
    }
} // norm_random

void new_beads(float *const v, const float *const sqvar)
// v are the new x-, y- or z-coordinates of the beads of a particle
{
    static float *vt = new float[np];

    for (int i = 1; i < np; ++i)
        vt[i] = sqvar[i]*norm_random();

    for (int i = 0; i < np; ++i) {
        v[i] = 0.0;
        for (int j = 1; j < np; ++j)
            v[i] += xmat[i][j]*vt[j];
    } // new_beads
C.3 Analyzation

anptpimclj analyzes the checkpoint file.

```c
#include <math.h>
#include <stdlib.h>
#include <iostream.h>
#include <fstream.h>
#include "float_rng.h"

// g l o b a l data; to be stored in the . d a t f i l e s

int n; // t o t a l n u m b e r o f p a r t i c l e s
int nA; // n u m b e r o f p a r t i c l e s o f s u b s t a n c e A
int np; // T r o t t e r n u m b e r ( b e a d s / p a r t i c l e )

float epsA; // L J−epsilon ( i n k J / m o l ) f o r s u b s t a n c e A
float sigA; // L J−sigma ( i n n m ) f o r s u b s t a n c e A
float mA; // p a r t i c l e m a s s ( i n k g ) f o r s u b s t a n c e A
float epsB; // L J−epsilon ( i n k J / m o l ) f o r s u b s t a n c e B
float sigB; // L J−sigma ( i n n m ) f o r s u b s t a n c e B
float mB; // p a r t i c l e m a s s ( i n k g ) f o r s u b s t a n c e B

float p; // p r e s s u r e
float t; // t e m p e r a t u r e
float v; // v o l u m e

float disp; // p a r t i c l e d i s p l a c e m e n t

float dvv; // v o l u m e c h a n g e p e r v o l u m e d v / v
float pvch; // p r o b a b i l i t y f o r v o l u m e c h a n g e

float pswp; // p r o b a b i l i t y f o r s w a p p i n g o f p a r t i c l e s
```
C.3 Analyzation

float dr; // channel width for pair-correlation
int ndr; // number of channels in the histogramm

long nt; // "time"
int ntskip; // thinning factor
int ntprint; // print control variables every ntprint*ntskip passes
int ntjob; // ntjob*ntskip passes

float** x; // x-coordinates for the n particles
float** y; // y-coordinates for the n particles
float** z; // z-coordinates for the n particles

float_rng the_rng; // the RNG

double accrpA; // accumulated acceptance ratio for particle moves for subst. A
double accrpB; // accumulated acceptance ratio for particle moves for subst. B
double accrv; // accumulated acceptance ratio for volume moves
double accrs; // accumulated acceptance ratio for particle swaps
double arho; // accumulated density
double ak; // accumulated kinetic energy for mixture
double auAA; // accumulated potential energy for interaction A<>A
double auAB; // accumulated potential energy for interaction A<>B
double auBB; // accumulated potential energy for interaction B<>B
double ah; // accumulated enthalpy for mixture
double av; // accumulated volume
double ar2A; // accumulated r**2 for substance A
double ar2B; // accumulated r**2 for substance B

double* agAA; // accumulated pair-correlation function for distance A<>A
double* agAB; // accumulated pair-correlation function for distance A<>B
double* agBB; // accumulated pair-correlation function for distance B<>B

// global data; not to be stored in the .dat files

const float au_c = 1.66053873e-27; // atomic mass unit in kg
const float NA_c = 6.02214199e23; // Avogadro's constant in 1/mol
const float kB_c = 1.3806503e-23; // Boltzmann's constant in J/K
const float R_c = 8.314472e-3; // molar gas constant in kJ/(mol K)

// inline functions

template<class T> inline T sqr(const T x)
{
    return x*x;
} // sqr(T)

template<class T> inline T cube(const T x)
{
    return x*x*x;
}
C The Code

} // cube(T)

// main

int main()
{
    char fname[100];
cout << "in file [out.dat]:" ;
cin getline(fname, 100, '\n');
if ( strcmp(fname,"" ) == 0)
    strcpy(fname, "out.dat");

ifstream fin(fname);

fin >> n >> nA >> np >> nt
    >> ntjob >> ntprint >> ntskip
    >> the_rng ;

    // check for zero-length run

if ( nt <= 0 ) {
    cout << "apimc:empty file." << endl ;
    fin.close();
    exit(1);
}

    // simulation parameters

fin >> disp >> dr
    >> epsA >> sigA >> mA
    >> epsB >> sigB >> mB
    >> p >> t >> v
    >> dvv >> pvch >> pswp ;

    // allocate arrays

float c = cbrt(v);

x = new (float*)[n];
y = new (float*)[n];
z = new (float*)[n];
for (int i = 0; i < n; ++i) { // np beads per particle
    x[i] = new float[np];
y[i] = new float[np];
z[i] = new float[np];
}

    // positions, accumulated averages & g(r)
for (int i = 0; i < n; ++i)
  for (int k = 0; k < np; ++k)
    fin >> x[i][k];
for (int i = 0; i < n; ++i)
  for (int k = 0; k < np; ++k)
    fin >> y[i][k];
for (int i = 0; i < n; ++i)
  for (int k = 0; k < np; ++k)
    fin >> z[i][k];
fin >> accrpA >> accrpB >> accrv >> acers
  >> arho
  >> ak
  >> auAA >> auAB >> auBB
  >> ah
  >> av
  >> ar2A >> ar2B;

// g(r)
fin >> ndr;
agAA = new double[ndr];
agAB = new double[ndr];
agBB = new double[ndr];
for (int i = 0; i < ndr; ++i)
  fin >> agAA[i] >> agAB[i] >> agBB[i];

// print results: simulation parameters
cout << "\n" << n << endl;
cout << "n\nA==" << nA << ')" << endl;
cout << "np==" << np << endl;
cout << "A:eps/k==" << epsA/Rc << 'K
" << endl;
cout << "A:sig==" << sigA << "nm" << endl;
cout << "A:m==" << mA/auA << "kg" << endl;
cout << "B:eps/k==" << epsB/Rc << 'K
" << endl;
cout << "B:sig==" << sigB << "nm" << endl;
cout << "B:m==" << mB/auA << "kg" << endl;
cout << "p==" << p/101325.0 << 'Pa" << endl;
cout << "t==" << t << 'K" << endl;
C The Code

cout << "v == NA_c/(1.0e21*n)*v << "ml/mol(" 
<< v << "nm**3")" << endl;
cout << "disp == disp << "nm" << endl;
cout << "dv/v == dvv << endl;
cout << "pvch == pvch << endl;
cout << "pswp == pswp << endl;
cout << "ndr == ndr << endl;
// averages

cout << endl;
cout << "nt == nt << "(1/1e21*n)*nt << "nm")" << endl;
cout << "acrpA == accrpA/nt << endl;
cout << "acrpB == accrpB/nt << endl;
cout << "acrv/nt == accrv/nt << endl;
cout << "acrs/nt == accrs/nt << endl;
cout << "<r*2> == sqrt(ar2A/nt) << "pm(" 
<< sqrt(ar2A/nt) << "nm")" << endl;
cout << "<r*2> == sqrt(ar2B/nt) << "pm(" 
<< sqrt(ar2B/nt) << "nm")" << endl;
cout << "<rho> == 1.0e24/NA_c*arho/nt << "mol/l (" 
<< arho/nt << "1/m**3")" << endl;
cout << "<k>/n == 1.0e3*ak/nt << "kJ/mol" << endl;
cout << "<\Delta u>/n == au/nt << "kJ/mol" << endl;
cout << "<\Delta u>/n == auAB/nt << "kJ/mol" << endl;
cout << "<\Delta u>/n == auBB/nt << "kJ/mol" << endl;
cout << "<\Delta u>/n == (auAA+auAB+auBB)/nt << "kJ/mol" << endl;
cout << "ah/nt == "kJ/mol" << endl;
cout << "cv == NA_c/(1.0e21*n)*av/nt << "ml/mol(" 
<< av/nt << "nm**3")" << endl;

// write g(r) ?
cout << endl;
cout << "Write g(r)? [y]: Keywords";

char copy, ch;
cin >> (copy);
if (copy != `'y'`) {
    cin.putback(copy);
    cin >> (copy);
} else {
    if (copy == `'y'` || copy == `'Y'`) {
        // Code for writing g(r)
    }
}
C.3 Analyzation

float pi = 4.0*atan(1.0);
float fact = 2.0*pi*arho/(3.0*n);
float fAA = fact*sqr(nA)*np;
float fAB = 2.0*fact*nA*(n-nA)*np;
float fBB = fact*sqr(n-nA)*np;

ofstream foutAA ("gAA. dat");
ofstream foutAB ("gAB. dat");
ofstream foutBB ("gBB. dat");
for (int i = 0; i < ndr; ++i) {
    float r1 = i*dr;
    float r2 = (i+1)*dr;
    float r = 0.5*(r1+r2);
    float dr3 = cube(r2)-cube(r1);
    foutAA << r << ' ' << agAA[i]/(fAA*dr3) << endl;
    foutAB << r << ' ' << agAB[i]/(fAB*dr3) << endl;
    foutBB << r << ' ' << agBB[i]/(fBB*dr3) << endl;
}
foutAA.close();
foutAB.close();
foutBB.close();

return 0;
} // main
C The Code
Bibliography


Bibliography


Bibliography


Bibliography


Index

asymmetric rule, 36, 37
Avogadro’s number, xiii

bi-dual space, 4
bijective, 5
BKL algorithm, 37
Boltzmann statistics, 23, 56
Bose statistics, 56
Box-Muller method, 108
bra, 4

cardinality, 5
Cauchy sequence, 4
  strong, 4, 6
  weak, 4, 6
Cauchy-Schwarz inequality, 4
classical isomorphism, 24, 49
classical mechanics, xiii, 11, 31
closure, 4
closure relation, 5
commutator, 9
complete orthonormal system (CONS), see Hilbert base
complete space, 4
concave function, 97
conditional probability, 34
configurational integral, 41
conjugated
  momentum, 11
convergence
  strong, 4, 7
  weak, 4, 7
convex function, 97
convex set, 97
cutoff distance, 44
dense
  domain, 7, 8
  set, 4
  subspace, 7
density in phase, see phase density
density matrix, 20–22
density operator, 10, 20
detailed balance, 35, 36
dimension, 5, 32
discrete Fourier transform (DFT), 50
distance, 4
dual space, 4, 7
energy
  kinetic, 12, 15, 17, 24, 73, 95
  potential, 12, 41, 42, 44, 46, 48, 73, 82, 95
ensemble
  canonical (NVT), 12, 13, 15–17, 20, 24,
  26, 27, 40–42, 46, 47, 49
  isothermal-isobaric (NpT), 15–17, 24,
  26, 44, 45, 47, 48, 95
  microcanonical (NVE), 25
enthalpy, 26
entropy, 25, 26
equipartition theorem, 15, 17, 24, 62, 73
ergodicity, 36
expectation value, 32
free energy, 26, 27
free enthalpy, 26, 27
functional, 3, 4
  continuous, 3
  linear, 3, 4
Gibbs potential, see free enthalpy
Hamilton function, 101
Hamilton operator, 20
Hamiltonian, 12–14, 16, 20, 101
Hamiltonian mechanics, 98–101
Helmholtz potential, see free energy
Hilbert basis, 4, 5, 21
Hilbert space, 3–10

importance sampling, 33
injective, 5, 7
inner energy, 25–27
inner product, 3, 4, 9, 10
ket, 3–5, 9, 10
Kronecker delta, 4
Lagrange function, 101
Laplace transformation, 103
Legendre duality, see Legendre transformation
Legendre transformation, 26, 27, 98–103
Legendre-Fenchel duality, 103
Lennard-Jones diameter, 39
Lennard-Jones potential, 39, 40, 44, 56
Lennard-Jones units, 44, 51
linear combination, 10
Lorentz-Berthelot combining rule, 81

macro-state, 11
macroscopic, 31
Markov chain, 34, 35
Markov matrix, 34
master equation, 34
matrix element, 9
mean value, 32
metric, 4
Metropolis method, 33, 34, 36, 37, 41
micro-state, 10, 11, 13, 14
Molecular Dynamics (MD), 31
Monte Carlo (MC), 31, 32, 37, 39, 41–43, 47–49

norm, 4, 5, 9
observable, 10, 11, 14, 15, 17, 20, 24, 31, 40, 41, 45, 47, 105
operator, 5–7, 9, 10, 19
adjoint, 7, 8
bounded, 5–8
closed, 6–8
compact, 7
continuous, 6, 7
strong, 6
weak, 6
hermitian, 8
inverse, 5, 8
isometric, 8
linear, 5–9
linear-bounded, 7
norm of, 5
self-adjoint, 6, 8, 10, 19
symmetric, 8
unbounded, 5, 7, 8
unitary, 6, 8
orthonormality relation, 4

pair-correlation function, 56, 63–67, 70, 72, 76–78, 80, 86, 90, 91
particle move, 41, 42, 45, 46, 48, 82
particle swap, 41, 47, 82
partition function
  canonical, 13–15, 20, 22, 23, 26, 62
  isothermal-isobaric, 15, 16, 26, 27, 46
  microcanonical, 25
pass, 41, 46
path-integral, 21
Path-Integral Monte Carlo (PIMC), 49, 56, 95
phase density, 11
phase space, 11, 15, 31, 45, 46
physico-chemical units, 44
probability amplitude, 10
probability density, 14, 15, 17, 32–34, 37, 40, 41, 45, 46, 51
probability distribution, 31, 34, 35
projector, 5
pure state, 10
quantum mechanics, 3–11, 62, 63, 82, 95
radial distribution function, see pair-correlation function
radius of gyration, 59, 62, 73, 82
random number, 108
random variable, 32, 37
ray, 9
rejection method, 33

scalar product, see inner product
scaled coordinates, 15, 45, 47
Schrödinger equation, 10
separability, 4
separable, 4
sesqui-linear form, 3
SI units, 44
space, 5
spectrum, 10
state vector, 3, 10
statistical operator, see density operator
superposition principle, 10
surjective, 5
symmetric rule, 36

tail correction, 44
theorem of
  Pythagoras, 4
  Riesz, 3
thermal de Broglie wavelength, 13, 14
thermal equilibrium, 25
thermodynamic limit, 43
thermodynamics, 25–27, 101–103
topology, 4
trace, 9
transformation method, 33
transition probabilities, 34–37
trial move, 36, 37, 41, 42, 46–48
triangle inequality, 4
Trotter number, 19, 21, 23, 56–59, 62–67, 72, 73, 76, 78

van der Waals force, 39
vector, 3, 4, 9
vector space, 3, 10
volume move, 44, 46, 48

Young’s inequalities, 100, 102
Danksagung

Zuvorderst bedanke ich mich bei meiner Mutter Eva Pfeiffer, die mir trotz aller Widrigkeiten das Studium erst ermöglicht hat.

Meinem Betreuer Prof. Martin Neumann danke ich für die interessante Aufgabenstellung und für die gute Zusammenarbeit. Prof. Hans Kuzmany danke ich, daß ich während der Mitarbeit in seiner Arbeitsgruppe an meiner Diplomarbeit weiterarbeiten konnte.

Curriculum Vitae

Personal Data
3 May 1974 born as the son of Rudolf and Eva Pfeiffer in Vienna, Austria

Education
1980–1984 Volksschule (primary school) in Vienna, Austria
1984–1988 AHS Unterstufe (first four years of secondary school) in Vienna, Austria
1988–1993 HTL (technical high-school) for Electrical Engineering (Energy Engineering and Power Electronics) at HTL Wien IV in Vienna, Austria
9 June 1993 Matura (graduation) at HTL Wien IV
10/1993–01/2001 Physics studies at the University of Vienna, Austria

Compulsory Military Service
11/1999–06/2000 Military scientific expert at the NBC-Defense School in Vienna, Austria