Wolfgang Schattke, Ricardo Díez Muiño

Quantum Monte Carlo Programming

for Atoms, Molecules, Clusters, and Solids
Wolfgang Schattke and
Ricardo Diez Muiño

Quantum Monte Carlo
Programming
Related Titles

Waser, R. (ed.)
Nanoelectronics and Information Technology
Advanced Electronic Materials and Novel Devices
2012
ISBN: 978-3-527-40927-3

Reimers, J. R.
Computational Methods for Large Systems
Electronic Structure Approaches for Biotechnology and Nanotechnology
2011
ISBN: 978-0-470-48788-4

Advanced Calculations for Defects in Materials
Electronic Structure Methods
2011
ISBN: 978-3-527-41024-8

Kroese, D. P., Taimre, T., Botev, Z. I.
Handbook of Monte Carlo Methods
2011
ISBN: 978-0-470-17793-8

Allinger, N. L.
Molecular Structure
Understanding Steric and Electronic Effects from Molecular Mechanics
2010
ISBN: 978-0-470-19557-4

Owens, F. J., Poole, Jr., C. P.
The Physics and Chemistry of Nanosolids
2008
ISBN: 978-0-470-06740-6

Reinhard, P.-G., Suraud, E.
Introduction to Cluster Dynamics
2004
ISBN: 978-3-527-40345-5

Landau, R. H., Páez, M. J.
Computational Physics
Problem Solving with Computers
1997
Wolfgang Schattke and Ricardo Díez Muiño

Quantum Monte Carlo Programming

for Atoms, Molecules, Clusters, and Solids
Contents

Preface  IX

1  A First Monte Carlo Example 1
  1.1 Energy of Interacting Classical Gas 1
  1.1.1 Classical Many-Particle Statistics and Some Thermodynamics 2
  1.1.2 How to Sample the Particle Density? 18

2  Variational Quantum Monte Carlo for a One-Electron System 23

3  Two Electrons with Two Adiabatically Decoupled Nuclei: Hydrogen Molecule 39
  3.1 Theoretical Description of the System 39
  3.2 Numerical Results of Moderate Accuracy 42
  3.3 Controlling the Accuracy 46
  3.4 Details of Numerical Program 53

4  Three Electrons: Lithium Atom 61
  4.1 More Electrons, More Problems: Particle and Spin Symmetry 63
  4.1.1 Antisymmetry and Decomposition of the Many-Body Wave Function 63
  4.1.2 Three-Electron Wave Function 65
  4.1.3 General Wave Function 67
  4.1.4 Relaxing Symmetry of Total Spin 70
  4.2 Electron Orbitals for the Slater Determinant 71
  4.3 Slater Determinants: Evaluation and Update 76
  4.4 Some Important Observables in Atoms? 82
  4.4.1 The Module “observables” 87
  4.5 Statistical Accuracy 91
  4.6 Ground State Results 93
  4.6.1 Results for Lithium Atom 93
  4.6.2 Code of Main Program, Modules of Variables, of Statistic, of Jastrow Factor, and of Output 103
  4.7 Optimization? 115

5  Many-Electron Confined Systems 121
  5.1 Model Systems with Few Electrons 121
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.2</td>
<td>Orthorhombic Quantum Dot</td>
<td>122</td>
</tr>
<tr>
<td>5.2.1</td>
<td>Confined Single-Particle Wave Functions</td>
<td>122</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Details of Program</td>
<td>123</td>
</tr>
<tr>
<td>5.2.3</td>
<td>Energy and Radial Density</td>
<td>125</td>
</tr>
<tr>
<td>5.2.4</td>
<td>Pair-Correlation Function</td>
<td>131</td>
</tr>
<tr>
<td>5.2.5</td>
<td>Program of the Pair-Correlation Function</td>
<td>134</td>
</tr>
<tr>
<td>5.3</td>
<td>Spherical Quantum Dot</td>
<td>136</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Fundamentals of DFT</td>
<td>137</td>
</tr>
<tr>
<td>5.3.2</td>
<td>DFT Calculation of the Jellium Cluster: Methodology</td>
<td>138</td>
</tr>
<tr>
<td>5.3.3</td>
<td>QMC Calculation of the Jellium Cluster: Methodology</td>
<td>140</td>
</tr>
<tr>
<td>5.3.4</td>
<td>QMC Code for the Calculation of Jellium Clusters</td>
<td>141</td>
</tr>
<tr>
<td>5.3.5</td>
<td>Comparison between DFT and QMC Calculations of Jellium Clusters</td>
<td>142</td>
</tr>
<tr>
<td>6</td>
<td>Many-Electron Atomic Aggregates: Lithium Cluster</td>
<td>147</td>
</tr>
<tr>
<td>6.1</td>
<td>Clusters and Nanophysics</td>
<td>147</td>
</tr>
<tr>
<td>6.2</td>
<td>Cubic BCC Arrangement of Lithium Atoms</td>
<td>150</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Structure of the Main Program</td>
<td>150</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Single-Electron Wave Functions and Structure of the Determinant</td>
<td>150</td>
</tr>
<tr>
<td>6.2.3</td>
<td>Geometric Setting of the Cluster</td>
<td>153</td>
</tr>
<tr>
<td>6.2.4</td>
<td>Changes in the Program</td>
<td>156</td>
</tr>
<tr>
<td>6.3</td>
<td>The Cluster: Intermediate between Atom and Solid</td>
<td>163</td>
</tr>
<tr>
<td>6.3.1</td>
<td>$1 \times 1 \times 1$ Cluster: Li$_2$</td>
<td>164</td>
</tr>
<tr>
<td>6.3.2</td>
<td>$2 \times 2 \times 2$ Cluster</td>
<td>167</td>
</tr>
<tr>
<td>6.3.3</td>
<td>$3 \times 3 \times 3$ Cluster</td>
<td>172</td>
</tr>
<tr>
<td>6.3.4</td>
<td>$4 \times 4 \times 4$ Cluster</td>
<td>174</td>
</tr>
<tr>
<td>6.3.5</td>
<td>Cluster Size</td>
<td>178</td>
</tr>
<tr>
<td>7</td>
<td>Infinite Number of Electrons: Lithium Solid</td>
<td>181</td>
</tr>
<tr>
<td>7.1</td>
<td>Infinite Lattice</td>
<td>183</td>
</tr>
<tr>
<td>7.1.1</td>
<td>The Lattices</td>
<td>183</td>
</tr>
<tr>
<td>7.1.2</td>
<td>Structure of the Electrostatic Potential</td>
<td>186</td>
</tr>
<tr>
<td>7.1.3</td>
<td>Ewald Summation and Tabulation</td>
<td>191</td>
</tr>
<tr>
<td>7.1.4</td>
<td>Finite-Size Effects</td>
<td>204</td>
</tr>
<tr>
<td>7.2</td>
<td>Wave Function</td>
<td>208</td>
</tr>
<tr>
<td>7.2.1</td>
<td>Linear Combination of Atomic Orbitals</td>
<td>208</td>
</tr>
<tr>
<td>7.2.2</td>
<td>Plane Waves</td>
<td>210</td>
</tr>
<tr>
<td>7.3</td>
<td>Jastrow Factor</td>
<td>212</td>
</tr>
<tr>
<td>7.3.1</td>
<td>Standard Choice</td>
<td>213</td>
</tr>
<tr>
<td>7.3.2</td>
<td>Principal Ideas and Extensions</td>
<td>215</td>
</tr>
<tr>
<td>7.4</td>
<td>Results for the $3 \times 3 \times 3$ and $4 \times 4 \times 4$ Superlattice Solid</td>
<td>216</td>
</tr>
<tr>
<td>8</td>
<td>Diffusion Quantum Monte Carlo (DQMC)</td>
<td>223</td>
</tr>
<tr>
<td>8.1</td>
<td>Towards a First DQMC Program</td>
<td>224</td>
</tr>
<tr>
<td>8.1.1</td>
<td>Relating Schrödinger Equation to Diffusion</td>
<td>224</td>
</tr>
<tr>
<td>8.1.2</td>
<td>Generate Gaussian Random Numbers</td>
<td>228</td>
</tr>
</tbody>
</table>
8.1.3 Application 229
8.1.3.1 Harmonic Oscillator 229
8.2 Conclusion 235

9 Epilogue 237

Appendix 239
A.1 The Interacting Classical Gas: High Temperature Asymptotics 239
A.2 Pseudorandom Number Generators 241
A.3 Some Generalization of the Jastrow Factor 247
A.4 Series Expansion 249
A.5 Wave Function Symmetry and Spin 257
A.5.1 Four Electrons 257
A.6 Infinite Lattice: Ewald Summation 259
A.7 Lattice Sums: Calculation 263

References 269

Index 273
Preface

The reader might be inclined not to read the preface when starting with the book, but rather at a later time when laziness or leisure leaves time for it. In the worst case, the reader might come back to the preface angered by some lack of understanding or, quite the opposite, angered by reading some undergraduate simplistic explanations. By consulting the preface, the reader is asking the authors about their goals in writing the book.

The main goal is declared by the book’s title, nevertheless with some restrictions in mind.

The following publication is settled somewhere between a textbook and a computer code manual. Its level is perhaps too specialized for a textbook and too broad for a manual. A positive comment would be that its content includes rather practical advice on what is usually described in a theoretical textbook, as well as presenting in more detail the physical understanding of what the manual of a code promises as a result. Dangling between these two extremes the authors could not decide where to place the book exactly, so they decided to take the risk of sharing the common ground in both.

Of course, one purpose was to make it more reader-friendly than a scientific paper, or a review article. However, reviews such as that of Foulkes, Mitas, Needs, and Rajagopal for example, represent invaluable sources for extended studies [1]. The path to fulfilling the purpose of a “friendly” book was led only by the authors’ own experience and will differ from that of others. In other words, neither the authors attended courses on “How to Write Pedagogically Good Books” nor did they read such literature.

Of course, the reader does not expect a manual coming with a scientific code which reduces to “read the input file explanations and then go on.” Therefore, instead of presenting just one code that could cover the general field, the authors decided to break up the program into pieces, each of them devoted to one of a few leading examples.

A pedagogic, but time and space-consuming possibility would have been to develop the codes step by step, to let the reader run into the many traps of programming errors with exercises and solutions. That could fill many volumes. We tried not to expand the volume beyond acceptable limits, but to keep enough material so
that the reader could start and develop from it his or her own specific programs. Therefore, we gave up on the textbook idea.

At the very first concept of this book we thought of presenting the code collected from the PhD theses of Eckstein and Bahnseh, who completed their work within the group of one of the authors (WS). It soon became clear that we would easily run into one of the difficulties cited above which we wanted to avoid. Therefore, we decided to program from scratch. In this way, we were also free to present our way of understanding the codes. In addition, we take on full responsibility of errors, not attributing them to any other source.

However, the number of mistakes unveiled and additionally those still hidden is embarrassing. Though some of the latter might be useful to track the path the code developed, they are not on purpose, we assure that. We present the code as it developed after testing and correcting as usual. Our main programming style, if we had any, was to render the code to be easily changed. This can be taken as an excuse for the lack of beauty and the lack of program efficiency. Both aspects and perspectives will be evaluated by the community differently with changing time, changing compilers, and changing computational facilities. To keep the work along the course of finding the pleasure in writing, we must admit deficiencies which we are now blamed for. We hope that the pleasure of eventually acquiring successful access to the quantum Monte Carlo scheme might outweigh the shortcomings from the reader’s point of view as well.

Thus, the book is not written to deliver an optimized program code. These codes exist and their development is left to another branch of science. Instead, we wanted to show some aspects of the vast and beautiful possibilities of the quantum Monte Carlo (QMC) method and to attract and maybe seduce the reader to devote his or her interest to this subject. We also want to touch on the various possibilities of choices of computing schemes connected with the method. The material presented here is by no way complete, and the general scientific development is not treated completely either. Some approaches are tentative and should be improved, some are clumsy and might be smoothed. Some parts are still under discussion.

After these atmospheric remarks, let us summarize the main topics that we included and some of those excluded from the content. We almost entirely focused on the variational quantum Monte Carlo (VQMC) scheme. The diffusion Monte Carlo (DMC) topic only covers a rather trivial example, the harmonic oscillator. There is another large branch of quantum Monte Carlo calculations for electron systems that we entirely omit here. It is based on the path integral with explicit fermion statistics. Relying on large computing resources it is used in a model-like manner for example for strong-coupling systems but rarely applied ab-initio to systems of material science.

VQMC is usually considered as the poor man’s version of QMC primarily because its theoretical concept is simple. One can refrain from the heavy complex machinery, which is hidden in the depths of quantum statistics, and calculate only the energy expectation value by a multidimensional integral and minimize the latter with respect to the parameters present in the wave function ansatz. The integral itself is computed with statistically chosen points of support, and that is the
stage at which some statistics enters. In particular, there is the belief in the central limit theorem stating that the procedure guarantees the reliability of those points which are drawn from a random walk. The problem lies in an adequate choice of a parameterized wave function. If there are many parameters, then one additionally has to utilize regression methods to obtain the best choice of them.

In contrast, the complexity of DMC is derived from the evolutionary scheme of a diffusion equation for the wave function, which should converge towards the true solution. Thus, one can dispense of an optimization procedure. Instead, one has to program the steps of the evolution, which is a combination of the separate actions of the kinetic and potential energy Hamiltonians on the actual wave function, to obtain the successive approximations. This combination as well as the generation of the random walkers, which mimic the wave function is less trivial. So we thought it important to explain this theoretical background and to show how it works with those easy going examples. To satisfy oneself with the role of being a theoretically poor man when devoting oneself to VQMC, one could imagine that DMC only replaces the optimization procedure of VQMC. One would also think that the physical insight lies in the choice of the functional shape of the many-body wave function rather than in obtaining its numerical representation as from DMC. Actually, in scientific calculations, one uses VQMC as the starting point and the rich man becomes again superior to the poor.

Presenting mainly VQMC in this volume, we proceed from simple examples such as the hydrogen atom, which has a known solution, to complicated ones such as the lithium solid. Being an infinite system, the latter presents a number of additional theoretical and numerical aspects which inflate the magnitude of the first example. Several intermediate steps are therefore inserted and explained: the hydrogen molecule, to deal with a two-electron system, going over to three electrons in the lithium atom, expanding to an arbitrary number of electrons when enclosed in a simple box potential or when assembled to an aggregate as a lithium cluster, to finally treating the three-dimensional periodic array of lithium atoms in a crystal. The two-electron system provides a first glance of particle symmetry in the wave function. The lithium atom stands for multiplicity and spin symmetry. Instead of localized orbitals, plane waves are utilized in a box, which also gives an opportunity to present the pair-correlation function. With the cluster of lithium atoms we discuss the role of a physical boundary that is important for the case of the infinite solid because of its shape-dependent energy terms, which only slowly converge with system size. The theory for the solid suffers from such terms resulting in an unacceptable slowing-down of convergence. Special remedies have to be discussed to this end, which complicate the program structure in addition to the routines already needed for a solid-state system.

The solid concludes the examples in the field of VQMC followed by the subject of DMC. Some detailed derivations are found at the end of the book in an appendix. The References cite suggestions for details and a deeper understanding of the material rather than exhaust the field or give honor to contributions for their historical importance.
One of the authors (WS) feels especially and gratefully obliged to the Donostia International Physics Center (DIPC) at the University of the Basque Country (UPV/EHU) for its long-lasting and generous hospitality. The time there rendered the development of this book an exciting experience and pleasure. In addition, these activities provided the opportunity to work for a period within the Ikerbasque community, which complemented the broad range of interests where he was embedded.

The other author (RDM) would like to thank the warm hospitality of the Christian-Albrechts-Universität of Kiel during part of the writing of this book. RDM is also extremely grateful to WS for teaching him how to maneuver in the intricate world of quantum Monte Carlo, and is definitely indebted to him for his patience and generosity.

Kiel
Donostia – San Sebastian
2012

Wolfgang Schattke
Ricardo Diez Muiño
A First Monte Carlo Example

What will be found in this chapter: We introduce randomness in a general way and we show how to deal with it in terms of probabilities and statistics. To illustrate the concepts, we start the book with an example based on classical physics, namely classical particles moving in a box. It is an example much simpler than those that involve quantum mechanics, but that already demonstrates the power of statistical physics and the deep insight offered by averaging magnitudes over many degrees of freedom.

1.1 Energy of Interacting Classical Gas

There are an overwhelming number of places in life where one is confronted with statistics: from a random binary even/odd decision, when picking petals off a flower to learn about the chances of being loved, to the refined probability distributions of health and age that life insurance companies use to estimate the premium [2].

Of course, the knowledge of how to treat ensembles of many elements appears to be much older and already shows the first traces of statistical insight. In ancient Asia Minor, for instance, the Hittites, who were strong in book-keeping, registered with eager interest the quantity of barley for their beer. No doubt that, for this purpose, they used some measurement pot instead of counting the grains in the bucket. Masters of cuneiform writing as they were, their alphabet would have had trouble counting huge numbers to enumerate the grains.

Statistical aspects emerged even more clearly in the past in the context of cryptography. In the early Islamic centuries, Arab scientists were very skilled in identifying the originality of texts which were attributed to Muhammad. To decode a text, the occurrences of single letters in a language can be counted. Such statistical analysis of languages was crucial to develop decoding algorithms able to solve outstanding problems of cryptology. For example, Roman military encoded their messages mixing the letters of the alphabet in a manner only known to the intended receiver, a procedure that resisted code-breaking for many centuries [3].

Mankind did not wait for the appearance of the Monte Carlo casinos to make their own statistical evaluations. By the way, those establishments for higher society provided numbers for the frequency of random events. These random numbers
were not only taken by a gambler to predict the outcome of subsequent throws but also by mathematicians to simulate any random process of unbiased events. The generation of random numbers by a mathematical algorithm is no trivial task. The quality of these so-called pseudorandom numbers strongly depends on the effort the algorithm invests. The need for random numbers is obvious for example in the numerical computation of high-dimensional integrations when the complexity of the integrand limits the evaluation to a small number of points. Then, choosing randomly distributed points represents a clear advantage over an equidistant grid if the distribution of the weight of points, uniform or nonuniform, can be guessed from the integrand itself. With the increased performance of computers, many fields embarked on this concept of integration. In its simplest case, the distribution can be chosen to be uniform, that is, nothing is known about it. In the main part of this book, however, we will show how the distribution of points in multidimensional integrals appearing in physics can be extracted from physics itself and its known statistical laws.

1.1.1 Classical Many-Particle Statistics and Some Thermodynamics

In this starting section we consider the relatively simple case of classical particles which move in a box and are described by their statistical behavior. This example avoids the complexity of quantum theory but shows already the statistical aspect of the general method, namely a scheme to average over many degrees of freedom.

Think of a rock concert in a huge hall filled with thousands of enthusiastic rapidly moving and hopping dancers. A significant pressure is exerted by the people onto the barrier of the stage. This pressure can be observed by how the dancers are spilled up to stage and how they are reflected in jumping back. And it is hot. So the hall, closed by doors and stage, is an example of a real gas with pressure and temperature, except that the particles are able to think (but who knows?).

The description of the behavior of many particles in terms of single-particle quantities such as exact positions or velocities quickly drops out of any feasible treatment when increasing the particle number. For more than one hundred years, the solution has been known (Boltzmann, Maxwell), focusing on practical grounds since nobody would be interested in the details. Nobody except the person himself/herself cares about the very elaborate moves another dancer of the concert hall is performing.

Moving now to physics, for a number of roughly $6 \times 10^{23}$ gas atoms per mole, it would be nearly impossible to identify all their positions, enumerate them, make a table, and communicate that information to someone else. Instead, one realizes that the three quantities, volume $V$ (pressure $p$), heat $Q$ (temperature $T$), and particle number $N$ (chemical potential $\mu$) or the bracketed ones, already provide a good description of the gas for a wide range of applications. Remember the historical Magdeburg half-spheres, where even the strength of two horses pulling in opposite directions was not enough to separate them. Or consider the weather forecast, by which people are more or less strongly affected: statements on the next day’s
weather are achieved by estimating the evolution of those thermodynamic quantities. The reason that such a description works (an optimistic view) is founded in statistical mechanics. The deviations from predictions, which question the reliability of weather forecasts, are influenced by turbulences whose treatment is formidable and involves statistical details not covered by the above-mentioned averaged quantities of thermodynamics. Nevertheless, the latter already yield a weather forecast we thankfully acknowledge.

Only the field of classical particles will be involved in the following statistics, leaving aside quantum properties which will be considered later. Here we focus for simplicity on a classical system. The equation of state of an ideal gas was significantly generalized as the van der Waals equation of the so-called real gases,

\[
(p - \frac{a}{V^2})(V - b) = N k_B T ,
\]

with Boltzmann constant \(k_B\). This equation is equivalent to that of an ideal gas for a volume reduced by the residual volume \(b\) of the molecular constituents and a pressure reduced by the inner pressure \(a/V^2\), which is exerted on the container wall by the particles’ repulsion. The \(1/V^2\) law and the constants \(a\) and \(b\) are either determined empirically or derived theoretically from statistics.

The subsequent considerations serve also as a test of the main statistical tool used in the variational quantum Monte Carlo (QMC) method besides the general common aspect. In fact, this kind of statistical investigation happened prior to the QMC development, showing at least their common roots. To be more specific we consider the derivation of the \(1/V^2\) law for the real gas. The idea behind introducing the Monte Carlo method in statistical mechanics is the multidimensional integration in statistical averages for many particles. For those interested in the relevant equations connecting thermodynamics and statistical mechanics for this example we give a short summary.

The key quantity is the free energy \(F(T, V) = -k_B T \ln Z(T, V)\) obtained from the classical partition function,

\[
Z(T, V) = \int_{\mathbb{V}^N} \frac{d(x_p)}{(2\pi \hbar)^3N} \exp \left\{ -\beta \sum_x \left[ \frac{p_i^2}{2m} + \frac{1}{2} \sum_{j \neq i} v_{ij}(x_i - x_j) \right] \right\}
\]

\[
= \left( \frac{1}{2\pi \hbar} \frac{2m\pi}{\beta} \right)^{3N/2} \int_{\mathbb{V}^N} \frac{d x^N}{(2\pi \hbar)^{3N/2}} \exp \left\{ -\beta \frac{1}{2} \sum_{i \neq j} v_{ij}(x_i - x_j) \right\}
\]

\[
= \left( \frac{1}{2\pi \hbar} \frac{2m\pi}{\beta} \right)^{3N/2} Z_{\text{pot}}(T, V) ,
\]

where \(\beta = 1/(k_B T)\) is related to the absolute temperature \(T\) and \(v_{ij}\) is the potential between two particles. The integration over the momenta \(p_i\) is carried out above
in closed form. The remaining integral abbreviated by $Z_{\text{pot}}$ has to be calculated numerically once a suitable interparticle potential is fixed. Before proceeding we state a few thermodynamical relations to connect the van der Waals equation (1.1) with the partition function. The partial derivative of the free energy with respect to volume, where temperature has to be kept constant, yields the pressure which is equated to that of the van der Waals state equation (1.1),

$$p = -\frac{\partial F(T, V)}{\partial V} \bigg|_{T=\text{const}} = k_B T \frac{\partial \ln Z_{\text{pot}}(T, V)}{\partial V} \bigg|_{T=\text{const}}$$

(1.5)

$$= \frac{k_B T}{Z_{\text{pot}}} \frac{\partial}{\partial V} \int_{V}^{N} \frac{d x^N}{(2\pi\hbar)^{N/2}} \exp \left[ -\beta \frac{1}{2} \sum_{i \neq j} v_{ij}(x_i - x_j) \right]$$

(1.6)

$$= N k_B T \frac{1}{V - b} + \frac{a}{V^2}.$$  

(1.7)

Expression (1.6) will be transformed into a form more suitable for a general Monte Carlo integration. To this end we differentiate the logarithm of the partition function $\ln Z_{\text{pot}}$ with respect to $-\beta$ which yields the average of the potential energy weighted by the Boltzmann probability density:

$$\frac{\partial \ln Z_{\text{pot}}(T, V)}{\partial (-\beta)} = Z_{\text{pot}}(T, V)^{-1} \int_{V}^{N} dx^N \frac{1}{2} \sum_{i \neq j} v_{ij}(x_i - x_j) \exp \left[ -\beta \frac{1}{2} \sum_{i \neq j} v_{ij}(x_i - x_j) \right]$$

(1.8)

$$=: U_{\text{pot}}.$$  

(1.9)

Reversely, we can reconstruct the partition function and furthermore the pressure from the average potential energy by integration, see (1.5), (1.8),

$$\ln Z_{\text{pot}}(\beta, V) = \ln Z_{\text{pot}}(\infty, V) - \int_{0}^{\beta} d\beta' U_{\text{pot}}(\beta', V),$$

(1.10)

$$p = k_B T \frac{\partial}{\partial V} \left[ N \ln \frac{V}{(\sqrt{2\pi\hbar})^3} - \int_{0}^{\beta} d\beta' U_{\text{pot}}(\beta', V) \right].$$

(1.11)

As a result, the relation between the pressure and the intended MC integration is given by an averaged potential energy $U_{\text{pot}}$, see (1.9),

$$p = \frac{k_B T N}{V} - k_B T \int_{0}^{\beta} d\beta' \frac{\partial U_{\text{pot}}}{\partial V}.$$  

(1.12)
Without special notation, in order to simplify the writing, we always keep constant the remaining variables in a partial differentiation or integration. Integrals which constitute an average as in (1.8) are within the scope of the Monte Carlo method we will dominantly use later. Here, it enables us to give an estimate for the state equation.

The potential energy will be fixed as a screened repulsive Coulomb potential, the so-called Yukawa potential. This is a convenient example for our main purpose which deals with electrons. Think here of a charged gas, though a real uncharged gas can be modeled similarly in the repulsive regime. For simplicity, we omit the hard core repulsion given by the finite extent of the molecules. As a consequence the constant $b$ in the van der Waals equation (1.1) must be set equal to zero. With screening length $\lambda$ and potential strength $v_0$ the Yukawa potential reads as

$$v_{ij} = \frac{v_0}{|r_i - r_j|} \exp \left( \frac{-|r_i - r_j|}{\lambda} \right), \quad (1.13)$$

using boldface types for three-dimensional vectors as the position vector $r_i$ of particle $i$. The $1/V^2$ volume dependence of the van der Waals pressure term is only an approximation of next lowest order to the ideal gas equation in an $1/V$ expansion which is called a virial expansion. Our MC simulation will not only display this term but the whole correction, being exact within this potential model and within the statistical error margin. More realistic calculations are based on the Lennard-Jones potential model and yield analytical results through an Ursell–Mayer cluster expansion. This is beyond the scope of this text though the interested reader could run a MC simulation with a more suitable interparticle potential.

With $b = 0$ we obtain from (1.1), (1.12) and finally integrating with respect to $V$

$$\frac{a}{V^2} = -k_B T \int_0^{\beta} d\beta' \frac{\partial U_{pot}}{\partial V}, \quad (1.14)$$

$$\frac{a}{V} = k_B T \int_0^{\beta} d\beta' U_{pot}, \quad (1.15)$$

where the integration constant vanishes at infinite volume.

We can approximately get some insight from the analytical point of view. At high temperature $\beta \to 0$ the multiple space integrations in (1.8) can be carried out exactly and reduce to volume averages over the potential energy of two particles, which is approximated by averaging $1/r_{12} \exp (-r_{12})$ over a spherical volume $V = L^3 = 4\pi/3 R^3$ of radius $R$ for a cubic box of edge length $L$,

$$U_{pot} \approx N(N - 1) v_0 \lambda^2 \frac{2\pi}{V} [1 - e^{-R/\lambda} (1 + R/\lambda)] \quad (1.16)$$

$$\approx N(N - 1) v_0 \lambda^2 \frac{2\pi}{V} \quad \text{for} \quad R/\lambda \gg 1 \quad (1.17)$$

$$\approx N(N - 1) v_0 \frac{3}{4} \left( \frac{4\pi}{3} \right)^{1/3} \frac{1}{V^{1/3}} \quad \text{for} \quad R/\lambda \ll 1 \quad (1.18)$$
The former case considers a screened interaction in the limit of a large volume. The latter applies to small volumes. Alternatively, if one likes to investigate the thermodynamic limit of infinite volume without screening, the screening has to be switched off first \( \lambda \to \infty \) before proceeding to the limit of large volume. The former gives the van der Waals law with

\[
a = 4\pi \frac{1}{2} N(N-1)\nu_0\lambda^2 =: N\nu_0\lambda^2\tilde{a}, \tag{1.19}
\]

where \( a \) is determined by the constants of the interaction potential. The factor \( 1/2N(N-1) \) counts the number of terms in the potential energy. It is compensated in the state equation by the square of the volume which is \( N \) times the volume per particle. Chemists prefer volume per mole, instead, by replacing \( Nk_B \) by the gas constant \( R \), see (1.1).

In view of the box geometry the approximation of (1.18) is too crude to compare with the subsequent exact numerical results. We need the exact asymptotical behavior. In Appendix A.1 the behavior for \( L/\lambda \ll 1 \) in the geometry which is used in the MC calculations is evaluated. It yields a single number 0.2353 to determine \( U_{pot} \), see (1.12), as \( U_{pot} = N(N-1)(4\nu_0/L)0.2353 \).

In order to use comfortable quantities for the MC simulation, instead of \( V, T, \) and \( U_{pot} \) we define dimensionless quantities

\[
v := \frac{V}{\lambda^3},
\]

\[
\alpha := \frac{\nu_0}{k_B T\lambda} = \frac{\nu_0\beta}{\lambda},
\]

\[
u(\alpha, v) := \frac{\lambda}{N\nu_0} U_{pot}(\beta, v),
\]

which leads with (1.14) to

\[
\frac{\tilde{a}}{v} = \frac{1}{\alpha} \int_0^\alpha d\alpha' u(\alpha', v). \tag{1.23}
\]

The above limits of \( U_{pot} \) control the numerical MC results obtained for the right-hand side of (1.23) in the case of high temperature, specifically

\[
\frac{1}{\alpha} \int_0^\alpha d\alpha' u(\alpha', v) = \frac{2\pi(N-1)}{v} \text{ for } \frac{L}{\lambda} \gg 1
\]

\[
= 4(N-1)\frac{0.2353}{v^{1/3}} \text{ for } \frac{L}{\lambda} \ll 1
\]

for \( T \to \infty \).

In the following we present and discuss the main parts of the program “therm95.f”. The program “therm95.f” uses the MC method to calculate the average energy of a system of particles confined in a cube under conditions of fixed
1.1 Energy of Interacting Classical Gas

The complete code is electronically available in the collection of programs.

A property is used which states that the statistical average of a quantity with a given probability density can be evaluated by summing a sequence of terms obtained from sequentially choosing points in density space according to their probability distribution. These terms are calculated as values of that quantity at the chosen points. Because of the sequential choice the terms can be considered as representing a fictitious time evolution of the quantity. The average over the statistical ensemble arises then as an average over evolution time where ergodicity would guarantee the equality of both averages.

Accordingly, the calculation of the average potential energy as specified in (1.9), (1.13) and (1.22) proceeds by simulating a random walk which samples the configuration space of the particles by a sequence of $M$ MC steps where in turn the position of every individual particle is submitted to a random change according to the Boltzmann probability. The “change according to probability” is achieved by proposing a step of random length and direction within a maximum step length and accepting it, if the probability of the new position is larger or equal to that of the old one. It is accepted also in the opposite case, that is, the ratio between the new and old probability is smaller than one, if the value of a one-dimensional random number drawn from a uniform distribution between zero and one lies below that ratio. Otherwise the proposed position is rejected and the actual particle stays at its old position. This reflects the true frequency of acceptances given by the probability distribution. The potential energy is simultaneously summed up adding its value at the actual position of each step. The sum divided by the number $M$ of steps represents at each stage a realization of a random energy variable, which is the normalized sum of sequential individual energies. According to the central limit theorem of probability theory such a sum converges under very general premises to a Gaussian distributed variable with a mean equal to the true average and a variance decreasing as $1/M$ to zero. Written by formulae the acceptance probability $p_{\text{step}}$ for a proposed step is obtained from the Boltzmann probability density of (1.9), $P_{\text{Boltzmann}}$, via

$$p_{\text{Boltzmann}} = Z_{\text{pot}}(T, V)^{-1} \exp \left[ -\beta \frac{1}{2} \sum_{i \neq j} v_{ij}(x_i - x_j) \right], \quad (1.26)$$

$$p_{\text{step}} = \min \left( \frac{P_{\text{Boltzmann}}^{\text{new}}}{P_{\text{Boltzmann}}^{\text{old}}}, 1 \right). \quad (1.27)$$

The factor $Z_{\text{pot}}(T, V)^{-1}$ represents the normalization of the probability. But we do not need to know it. It is a nice property of the so-called Metropolis algorithm described above which states that the procedure automatically guarantees normalization. For instance, if instead of summing the energy we sum an arbitrary constant, say $c$, then the sum divided by $M$ is equal to $c$, which demonstrates automatic normalization.

The need for random numbers occurs at two stages. First, one has to decide upon direction and length of a step in position space. To this aim one runs over the...
the three Cartesian directions and chooses every coordinate extent as a fraction of a maximum step length by drawing a random number uniformly distributed between zero and one. Second, one has to decide upon acceptance of a MC step in case of a smaller probability at the new particle position.

Throwing a die and letting statistical events decide one’s fate has been common to mankind from its earliest stages. From written testimonies of antiquity down to relics in the tombs of ancient Egypt, or of any region including the Far East, the dice always had a similar appearance, except that some were loaded for cheating. Thus, some statistical concepts must have been known since early times, though the art of gambling experienced a refinement in our ages with the construction of casinos as in Monte Carlo. In casinos a large reservoir of random numbers was developed, which could be utilized for statistical investigations instead of throwing the dice. Drawing random numbers has now become even more easy by the use of computers. We will present details of random number generators in a subsequent section, but here we use without discussion the built-in generator of the g95-Fortran compiler. The relevant code is contained in the “module random” of the program.

A dumb choice of starting positions, as it might occur for instance by an extremely improbable configuration could influence the subsequent positions for a number of steps and deteriorate the value of the sampled energy. To become independent of the initial conditions, one applies a first MC run without counting the energy, our so-called prerun, which otherwise has the same structure as the main run. Furthermore, we skip the formal specification part of the program as well as both outer loops, which run over the different values of two external parameters, the temperature parameter ALPHA and the cubic length parameter LENGTH. Programming of the output data does not need explanation either. Dots denote program parts left out from this representation.

C-----------------------------
program THERM
use random
use position
use output,only: FE1MAX,FE2MAX,FE3MAX,PRONAME,OUT3D
C NE = number of particles
implicit none
integer,parameter :: NXFA1=0,NXFA2=9
real(dp) :: X,Y,Z,STEPMAX,ALPHA,DALPHA,ALPHA0,ALPHA1,BETA
real(dp) :: DLENGTH,LENGTH0,LENGTH1,DUMMY
real(dp) :: RDIF,QUOT
real(dp) :: LOCEN,ERWEN,VAREN,ERWKIN,VARKIN,ERWPOT,VARPOT
real(dp) :: AVRHO(FE3MAX,FE3MAX,FE3MAX)
logical :: MCSCHRITT,SWIRHO
character(7) :: FELDNAM
C Local variables
integer :: n1,n2,i,k,ny,nz,M
real(dp) :: RAD1,RAD2,RADNUE,EK,EP,ranumb
real(dp) :: WORK1(NXFA2+1),WORK2(NXFA2+1),WORK
logical :: l
1.1 Energy of Interacting Classical Gas

C
PRONAME='thermod'
C Control output
    open(unit=35,file=PRONAME/_MC10.OUT,position='append',
         & status='unknown')
    write(35,*),'MC7 and MC6 have totally unbiased boundary condition'
    write (35,*),'totally unbiased is still biased; instead MC8 rejects'
    write (35,*),'any move outside the volume which is really unbiased'
C Contour plot output ALPHA and WAVEC for "xfarbe" tool
C on a NXFA*NXFA grid
    open(unit=38,file=PRONAME/erw_yukawa10.dat,position='append',
         & status='unknown')
    open(unit=39,file=PRONAME/var_yukawa10.dat,position='append',
         & status='unknown')
C Number of electrons and MC steps
    NE = 100
    IF (NE .GT. NEMAX) THEN
        write (*,*),'NE= ',NE,' larger than NEMAX= ',NEMAX
        STOP
    ENDIF
    MCPRE = 100000
    MCMAX = 1000000
    NDIV = 21   ! NDIV must be odd
    call INITRAN
C Start data:
    lparam1:do m=1,3
    lparam2:do n1 = 1,NXFA1+1
    lparam3:do n2 = 1,NXFA2+1
C Parameter ALPHA of inverse temperature and edge length LENGTH
    SWIRHO = .true.
    ALPHA0=0.50D0
    ALPHA1=10.00D0
    LENGTH0=+1.00D0
    LENGTH1=+50.00D0
    DALPHA=(ALPHA1-ALPHA0)/DBLE(NXFA1+1)
    DLENGTH=(LENGTH1-LENGTH0)/DBLE(NXFA2+1)
    ALPHA=ALPHA0+(n1-1)*DALPHA
C adjust LENGTH to logarithmic plot
    LENGTH=(0.01+(n2-1)*0.01)*10**m
C Maximum step width, Be careful: check with acceptance ratio!
    STEPMAX = 9.00D-2*LENGTH
C Random initial electron positions
    do k=1,NE
        do i=1,3
            call GENRAN(rannumb)
            RE(i,k) = LENGTH*(rannumb-0.5)
            RNEU(i,k) = RE(i,k)
        end do
    end do
    do i=1,NE
        VNEW(i) = 0.D0
        VDIF(i) = 0.D0

do k=1,NE
   IF (k .eq. i) cycle
   DISTNEU(1:3,i,k) = RNEU(1:3,i)-RNEU(1:3,k)
   DIST(1:3,i,k) = DISTNEU(1:3,i,k)
   DISTNEU(4,i,k) = dsqrt(sum (DISTNEU(1:3,i,k)**2))
   DIST(4,i,k) = DISTNEU(4,i,k)
   VNEW(i) = VNEW(i) + 1.D0/DISTNEU(4,i,k)*dexp(-DISTNEU(4,i,k))
end do
end do

C Counts the acceptance number
MCOUNT = 0

C Observables
RHO(1:NDIV,1:NDIV,1:NDIV) = 0.D0
AVRHO(1:NDIV,1:NDIV,1:NDIV) = 0.D0
LOCEN = 0.D0
ERWEN = 0.D0
VAREN = 0.D0
ERWKin = 0.D0
VARKIN = 0.D0
ERWPOT = 0.D0
VARPOT = 0.D0
l = .true.

C MC loop: prerun for thermalizing. Be careful: does not change a bad sampling of energy!!!
lrunpre:do IMC=2,MCPRE
   lelrunpre:do IE=1,NE
      do i=1,3
         C Shift position at random within +-STEPMAX/2
         call GENRAN(rannumb)
         RDIF = (rannumb-0.5)*STEPMAX
         RNEU(i,IE) = RE(i,IE)+RDIF
         C Reflect particle at wall if it crosses
         if (RNEU(i,IE) > 0.5D0*LENGTH)
            RNEU(i,IE)=LENGTH-RNEU(i,IE)
         if (RNEU(i,IE) < -0.5D0*LENGTH)
            RNEU(i,IE)=-LENGTH-RNEU(i,IE)
         C Scatter particle at a rough wall
         if (DABS(RNEU(i,IE)) > 0.5D0*LENGTH) THEN
            call GENRAN(rannumb)
            if (RNEU(i,IE) > 0.5D0*LENGTH)
               RNEU(i,IE)=LENGTH/2.D0-ABS(rannumb-0.5)*STEPMAX
            if (RNEU(i,IE) < -0.5D0*LENGTH)
               RNEU(i,IE)=-LENGTH/2.D0+ABS(rannumb-0.5)*STEPMAX
            if (RNEU(i,IE) < -0.5D0*LENGTH)
               RNEU(i,IE)=-LENGTH/2.D0+ABS(rannumb-0.5)*STEPMAX
            if (RNEU(i,IE) < -0.5D0*LENGTH)
               RNEU(i,IE)=-LENGTH/2.D0+ABS(rannumb-0.5)*STEPMAX
         C Apply strict cut-off boundary conditions
         if (DABS(RNEU(i,IE)) > 0.5D0*LENGTH) THEN
            MCSCHRITT = .false.
            RNEU(1:3,IE) = RE(1:3,IE)
            MCOUNT = MCOUNT + 1
            cycle lelrunpre
1.1 Energy of Interacting Classical Gas

end if
end do
C Calculate Boltzmann ratio with energy 0.5*sum_k 1/r_ik exp(-r_ik)
C without term k=i
  call JASEXP(VNEW(IE),VDIF(IE))
  QUOT = DEXP(-ALPHA*VDIF(IE))
C Test on acceptance
  if (QUOT .lt. 1) THEN
    MCSCHRITT = DBLE(ZBQLU01(DUMMY)) .LT. QUOT
    call GENRAN(rannumb)
    MCSCHRITT = DBLE(rannumb) < QUOT
  end if
  if (MCSCHRITT) then
    RE(1:3,IE) = RNEU(1:3,IE)
    MCOUNT = MCOUNT + 1
  else
    RNEU(1:3,IE) = RE(1:3,IE)
  end if
end do lelrunpre
end do lrunpre
write(35,*)'STEPMAX = ',STEPMAX
write(35,*)'prerun: MCPRE= ',MCPRE,' acc. ratio = ',
  100.*DBLE(MCOUNT)/DBLE(NE*MCPRE),' %'
MCOUNT = 0
C
C MC loop: main run after thermalizing
lrun:do IMC=2,MCMAX
  lelrun:do IE=1,NE
    do i=1,3
      C Shift position at random within +-STEPMAX/2
      call GENRAN(rannumb)
      RDIF = (rannumb-0.5)*STEPMAX
      RNEU(i,IE) = RE(i,IE)+RDIF
      CCC Reflect particle at wall if it crosses
      if (RNEU(i,IE) > 0.5D0*LENGTH) then
        RNEU(i,IE)=LENGTH-RNEU(i,IE)
      end if
      if (RNEU(i,IE) < -0.5D0*LENGTH) then
        RNEU(i,IE)=-LENGTH-RNEU(i,IE)
      end if
      CCC Scatter particle at a rough wall
      if (DABS(RNEU(i,IE)) .gt. 0.5D0*LENGTH) then
        call GENRAN(rannumb)
        if (RNEU(i,IE) > 0.5D0*LENGTH) then
          RNEU(i,IE)=LENGTH/2.0D0-ABS(ZBQLU01(DUMMY)-0.5)*STEPMAX
        end if
        if (RNEU(i,IE) < -0.5D0*LENGTH) then
          RNEU(i,IE)=-LENGTH/2.0D0+ABS(ZBQLU01(DUMMY)-0.5)*STEPMAX
        end if
        if (RNEU(i,IE) > 0.5D0+ABS(ZBQLU01(DUMMY)-0.5)) then
          RNEU(i,IE)=LENGTH/2.0D0-ABS(rannumb-0.5)*STEPMAX
        end if
        if (RNEU(i,IE) < -0.5D0+ABS(ZBQLU01(DUMMY)-0.5)) then
          RNEU(i,IE)=-LENGTH/2.0D0+ABS(rannumb-0.5)*STEPMAX
        end if
        if (RNEU(i,IE) > 0.5D0-ABS(rannumb-0.5)) then
          RNEU(i,IE)=LENGTH/2.0D0-ABS(rannumb-0.5)*STEPMAX
        end if
        if (RNEU(i,IE) < -0.5D0-ABS(rannumb-0.5)) then
          RNEU(i,IE)=-LENGTH/2.0D0+ABS(rannumb-0.5)*STEPMAX
        end if
      end if
    end do i
  end do IE
end do IMC
end do lrunpre
CC & RNEU(i,IE)=-LENGTH/2.D0+ABS(rannumb-0.5)*STEPMAX
CC & RNEU(i,IE)=-LENGTH*(1/2.D0-rannumb) ! totally unbiased
C Apply strict cut-off boundary conditions
  if (DABS(RNEU(i,IE)) > 0.5D0*LENGTH) THEN
    MCSCHRITT = .false.
    l = .false.
    exit
  end if
end do
if (l) then
  C Calculate Boltzmann ratio with energy 0.5*sum_k 1/r_ik exp(-r_ik)
  C without term k=i
  call JASEXP(VNEW(IE),VDIF(IE))
  QUOT = DEXP(-ALPHA*VDIF(IE))
  C Test on acceptance
  if (QUOT < 1) then
    call GENRAN(rannumb)
    MCSCHRITT = DBLE(rannumb) < QUOT
  else
    MCSCHRITT = .true.
  end if
  end if
  l = .true.
C Update of observables.
  if (MCSCHRITT) then
    RE(1:3,IE) = RNEU(1:3,IE)
    ep = VNEW(IE)
    MCOUNT = MCOUNT + 1
  else
    RNEU(1:3,IE) = RE(1:3,IE)
    ep = VNEW(IE) - VDIF(IE)
  end if
C E=0.5 sum_ik v_ik, sum i omitted because
C run averages over particles, so energy per particle is calculated
LOCEN = LOCEN + 0.5D0*ep
end do lelrun
C energy per particle
LOCEN = LOCEN/DBLE(NE)
if (IMC == 2) then !set start values
  ERWEN = LOCEN
  VAREN = 0.D0
end if
ERWEN = (IMC-1)/DBLE(IMC)*ERWEN+LOCEN/DBLE(IMC)
VAREN = (IMC-1)/DBLE(IMC)*VAREN +
& 1/DBLE(IMC-1)*(ERWEN-LOCEN)**2
LOCEN = 0.D0
if (SWIRHO) then
  C density
  call DENSITY(RHO)
  do nz=1,NDIV
    do ny=1,NDIV
      AVRHO(1:NDIV,ny,nz)=(IMC-1)/DBLE(IMC)*AVRHO(1:NDIV,ny,nz)
      & + RHO(1:NDIV,ny,nz)/DBLE(IMC)
 Modules “random”, “position”, and “output” are used, which specify the random number generator, the updating of the particle positions and their potential energy, and the output collection. The values for the number of particles, number of steps of the prerun and of the main MC run are chosen as \( N_E = 100 \), \( M_{CPR} = 10000 \), \( M_{CMA} = 100000 \) in this example. The maximum step width \( \text{STPMA} \) is taken as somewhat less than the actual length parameter \( \text{LENGTH} \), which denotes the edge length of the container cube. A random number is calculated by subroutine \( \text{GENRAN}(rannumb) \). The initial particle positions \( \text{RE}(1:3,1:N_E) \) are randomly chosen.

The arrays \( \text{DIST}(1:4,1:N_E,1:N_E) \) and \( \text{DISTNEW}(1:4,1:N_E,1:N_E) \) both store the interparticle distance vectors, the former for the old configuration and the latter for the updated one. At start both are identical. The fourth component stores its modulus. \( \text{VNEW}(1:N_E) \) and \( \text{VDIF}(1:N_E) \) store the updated potential and the difference between it and its nonupdated value of a particle in the field of the others, see (1.13). Initializing is completed with the initial setting of the counter \( \text{MCOUNT} \) which counts each accepted case of MC steps and with the setting of the energy
variables, that is, the actual potential energy LOCEN, its average ERWEN, and its variance VAREN.

Here, we describe only the main MC run which is started after the prerun, because the prerun is essentially the main run without the calculation of observables. The MC steps are counted by IMC and for every step a run over all NE particles is carried out with IE as actual particle index. The update process begins by proposing a jump of the actual particle to a new position which is obtained by increasing its three coordinates by randomly chosen increments within the interval \( \pm \text{STEPMAX}/2 \). RNEU(1:3,1:NE) stores the new position of that particle. Subsequently, the constraint for enclosing the particles in a cubic box is applied by randomly back scattering the particle if the new position crosses the container walls. Alternatively, one could think of applying reflecting boundary conditions or requiring a strict cut-off at the surface. The reflecting walls could mean nonergodic runs like the behavior of a billiard ball enclosed between ideally reflecting walls, which comes close to the case of a very dilute gas. It is left to the reader to test how this changes our results. We will come back to this point in Section 1.1.2, though a test with strict cut-off on the case of \( \alpha \neq 0.5 \) in Figure 1.1 does not change the graph.

To calculate the ratio QUOT between the probability after the jump and that before it, subroutine JASPEX is called which involves the module “position” and is explained in detail later in context with that module. Here, it suffices to state that the new potential VNEW(IE) and its difference VDIFF(IE) with respect to the old one is obtained by this routine. As only the position of the actual particle IE is changed at the actual jump, interparticle potential contributions between the actual particle only and all other particles are involved. The factor ALPHA in the exponent stands for the temperature, see (1.13) and (1.20). Then, as explained in the theory, the ratio QUOT is compared with 1 and the logical variable MCSCHRITT set to “true” if QUOT is larger than or equal to 1. Otherwise MCSCHRITT gets the value “true” in those cases where a random number drawn from \([0,1]\) lies below QUOT and “false” otherwise.

In the case that MCSCHRITT is “true” the position variable RE(3,NE) is updated to its new value and the acceptance counter is incremented. Otherwise the variable RNEU(3,NE) holding the new position is reset to the old position. The potential energy “ep” of the actual particle IE is given the actual value according to the new or old position by the last call to JASEXP, and added to the energy sum of previous particles. For the energy per particle we divide by NE and update the average energy ERWEN and its variance VAREN. The formulae for the latter quantities are addressed in the section on statistical properties. We have reached the end of the MC loop with label 100. The remaining part, not cited here, concerns the output. Note that we reserve channel 35 for control output in a file with ending MC.OUT. Channels 38, 39 direct the average energy ERWEN and its variance VAREN to the files therm_erw_yukawa.dat and therm_var_yukawa.dat, respectively, for appending the results of multiple MC runs.

It remains to discuss the subroutine JASEXP contained in the module “position” as found below.
1.1 Energy of Interacting Classical Gas

\[ u(\alpha, v) = \frac{4(N-1)0.2353}{L^3} \]

Figure 1.1 Van der Waals pressure correction to ideal gas equation of state according to (1.22) in logarithmic representation, \( u(\alpha, v) \) vs. \( L = \sqrt[3]{N} \). Note that here the edge length \( L \) of the cubic container volume is already scaled by the screening length \( \lambda \) of the interaction potential. Thick lines without symbols show analytical results \( 2\pi(N-1)/L^3 \) and \( 4(N-1)0.2353/L \) for high temperature \( \alpha \to 0 \) at both asymptotic limits \( v \to 1 \) and \( v \to 0 \), respectively; dots connected by thin solid lines refer to MC simulations with \( \alpha = 0.5 \) showing separate calculation sets with different maximum step widths \( \text{STEPMAX} = 0.001, 0.01, 0.1, \text{ and } 1.0 \) for lengths \( L \) between 0.1–1, 1–10, 10–50, and above 50, respectively, calculated at high accuracy with MC steps equal to \( 10^6 \) times number of particles \( N \). Lower accuracy with only \( 10^5 \) times number of particles is shown by open squares for comparison; lines with star symbols refer to different values of \( \alpha \) as indicated and lower accuracy \( (10^5) \), number of particles is \( N = 100 \) in all cases; inset shows an extract for very small volume.

```fortran
module position
C updates positions, their differences, two-particle potential and C its change for particle IE
implicit none
public :: JASEXP
integer,parameter,public :: NEMAX=1000,
& dp=selected_real_kind(2*precision(1.0))
integer,public :: NE,IE
real(kind=dp),public:: EMACH
real(kind=dp),dimension(NEMAX),public :: VNEW,VDIF
real(kind=dp),dimension(3,NEMAX),public :: RE,RENEW
real(kind=dp),dimension(4,NEMAX,NEMAX),public :: DIST,DISTNEW
contains
```

C
subroutine JASEXP(VN,VD)
C updates the distances from the active particle to all others
C and calculates potential energy
real(kind=dp),intent(out),dimension(NEMAX):: VN,VD
integer :: i,k,n
real(dp) :: work01,work02,work1,work2
work1=0.0_dp
work2=0.0_dp
IENEK:do k=1,NE
   if (k .eq. IE) then
      cycle
   end if
   work01=0.0_dp
   work02=0.0_dp
   do n=1,3
      DIST(n,IE,k)=RE(n,IE)-RE(n,k)
      DIST(n,k,IE)=-DIST(n,IE,k)
      DISTNEW(n,IE,k)=RENEW(n,IE)-RE(n,k)
      DISTNEW(n,k,IE)=-DISTNEW(n,IE,k)
      work01=work01+DIST(n,k,IE)**2
      work02=work02+DISTNEW(n,k,IE)**2
   end do
   work01 = dsqrt(work01)
   work02 = dsqrt(work02)
   if (work01 .lt. EMACH) work01 = 2.D0/3.D0*EMACH
   if (work02 .lt. EMACH) work02 = 2.D0/3.D0*EMACH
   DIST(4,IE,k) = work01
   DIST(4,k,IE) = work01
   DISTNEW(4,IE,k) = work02
   DISTNEW(4,k,IE) = work02
   work1=work1+1.D0/work02*dexp(-work02)-1.D0/work01*dexp(-work01)
   work2=work2+1.D0/work02*dexp(-work02)
end do IENEK
VD(IE)=work1
VN(IE)=work2
end subroutine JASEXP
end module position

The interparticle distances refer to the actual particle IE and concern their value with respect to the new DISTNEW(3,IE,NE) and to the old position DIST(3,IE,NE). With these distances the new potential energy VN(IE) and its difference VD(IE) from the old potential energy is determined according to (1.13) and transferred to the calling program. Note that the potential energy per particle must carry a factor 1/2 which is multiplied in the calling program. In contrast the probability ratio QUOT contains the total potential energy, which is twice that value because the particle IE appears twice in the particles’ double sum.

Figure 1.1 displays over several length decades how the exact result for high temperatures is approached by the MC simulation. The edge length of the volume is scaled by the screening length of the interaction potential such that at $\lambda$ the transition occurs between unscreened potential at small distances and fully screened potential at large distances. The plot shows a rapid convergence towards the van