

ms2: A molecular simulation tool for thermodynamic properties, new version release

Colin W. Glass^a, Steffen Reiser^b, Gábor Rutkai^c, Stephan Deublein^b, Andreas Köster^c, Gabriela Guevara-Carrion^c, Amer Wafai^a, Martin Horsch^b, Martin Bernreuther^a, Thorsten Windmann^c, Hans Hasse^b, Jadran Vrabec^{c,*}

^aHöchstleistungsrechenzentrum Universität Stuttgart (HLRS), 70550 Stuttgart, Germany

^bLehrstuhl für Thermodynamik, Universität Kaiserslautern, 67653 Kaiserslautern, Germany

^cLehrstuhl für Thermodynamik und Energietechnik, Universität Paderborn, 33098 Paderborn, Germany

Abstract

A new version release (2.0) of the molecular simulation tool *ms2* [S. Deublein et al., Comput. Phys. Commun. 182 (2011) 2350] is presented. Version 2.0 of *ms2* features a hybrid parallelization based on MPI and OpenMP for molecular dynamics simulation to achieve higher scalability. Furthermore, the formalism by Lustig [R. Lustig, Mol. Phys. 110 (2012) 3041] is implemented, allowing for a systematic sampling of Massieu potential derivatives in a single simulation run. Moreover, the Green-Kubo formalism is extended for the sampling of the electric conductivity and the residence time. To remove the restriction of the preceding version to electro-neutral molecules, Ewald summation is implemented to consider ionic long range interactions. Finally, the sampling of the radial distribution function is added.

Keywords: Molecular simulation, molecular dynamics, Monte Carlo, vapor-liquid equilibrium, transport properties, Massieu potential derivatives

1. Program summary

Manuscript: *ms2*: A molecular simulation tool for thermodynamic properties

Authors:

Colin W. Glass, Steffen Reiser, Gábor Rutkai, Stephan Deublein, Bernhard Eckl, Jürgen Stoll, Sergey V. Lishchuk, Andreas Köster, Gabriela Guevara-Carrion, Amer Wafai, Martin Horsch, Thorsten Merker, Martin Bernreuther, Thorsten Windmann, Hans Hasse, Jadran Vrabec (jadran.vrabec@upb.de)

Title of program: *ms2*

Operating system: Unix/Linux

Computer: The simulation program *ms2* is usable on a wide variety of platforms, from single processor machines to modern supercomputers.

*Corresponding author: Jadran Vrabec, Warburger Str. 100, 33098 Paderborn, Germany, Tel.: +49-5251/60-2421, Fax: +49-5251/60-3522, Email: jadran.vrabec@upb.de

Memory: *ms2* runs on single cores with 512 MB RAM. The memory demand rises with increasing number of cores used per node and increasing number of molecules.

Distribution format: tar.gz

Keywords: Molecular simulation, molecular dynamics, Monte Carlo, vapor-liquid equilibrium, transport properties, Massieu potential derivatives

Programming language: Fortran90

External: Message passing interface (MPI)

Classification: 7.7, 7.9, 12

Has the code been vectorized or parallelized: Yes: Message Passing Interface (MPI) protocol and OpenMP Scalability is up to 2000 cores.

Nature of problem: Calculation of application oriented thermodynamic properties for fluids consisting of rigid molecules: vapor-liquid equilibria of pure fluids and multi-component mixtures, thermal and caloric data as well as transport properties.

Method of solution: Molecular dynamics, Monte Carlo, various classical ensembles, grand equilibrium method, Green-Kubo formalism, Lustig formalism.

Restrictions: None. The system size is user-defined. Typical problems addressed by *ms2* can be solved by simulating systems containing typically 1'000 – 4'000 molecules.

Unusual Features: Auxiliary feature tools are available for creating input files, analyzing simulation results and visualizing molecular trajectories.

Additional comments: Sample makefiles for multiple operation platforms are provided.

Documentation: Documentation is provided with the installation package and is available at <http://www.ms-2.de>.

Typical running time: The running time of *ms2* depends on the specified problem, the system size and the number of processes used in the simulation. E.g. running four processes on a "Nehalem" processor, simulations calculating vapor-liquid equilibrium data take between two and 12 hours, calculating transport properties between six and 24 hours. Note that the examples given above stand for the total running time as there is no post-processing of any kind involved in property calculations.

Catalogue identifier of previous version: AEJF_v1_0

Journal Reference of previous version: Comput. Phys. Commun. 182 (2011) 2350

Reasons for new version: The source code was extended to introduce new features.

Summary of Revisions: The new features of Version 2.0 include: hybrid parallelization based on MPI and OpenMP for molecular dynamics simulation; Ewald summation for long range interactions; sampling of Massieu potential derivatives; extended Green-Kubo formalism for the sampling of the electric conductivity and the residence time; radial distribution function.

Does the new version supersede the previous version: Version 2.0 brings new features, but does not necessarily supersede the previous one.

2. Introduction

Molecular modeling and simulation is a technology central to many areas of research in academia and industry. With the advance of computing power, the scope of application scenarios for molecular simulation is widening, both in terms of complexity of a given simulation and in terms of high throughput. Nowadays, e.g. the predictive simulation of entire phase equilibrium diagrams has become feasible. However, in order to rely on simulation results, the methodology needs to be sound and the implementation must be thoroughly verified. In its first release [1], we have introduced the molecular simulation tool *ms2*. Results from *ms2* have been verified and the implementation was found to be robust and efficient.

As described in Section 3, in Version 2.0 of the simulation tool *ms2* the existing molecular dynamics (MD) MPI parallelization was hybridized with OpenMP, leading to an improved performance on multi-core processors. Furthermore, the new release offers a wider scope of accessible properties. In particular, *ms2* was extended to calculate Massieu potential derivatives in a systematic manner, cf. section 4. This augments the range of sampled properties significantly and, as was demonstrated in [2], it allows to straightforwardly develop competitive fundamental equations of state from a combination of experimental VLE data and molecular simulation results. Lastly, besides being now capable of simulating ionic substances, the time and memory demand for calculating transport properties was reduced significantly (section 5).

ms2 is freely available as an open source code for academic users at www.ms-2.de.

3. Hybrid MPI & OpenMP Parallelization

The molecular simulation tool *ms2* focuses on thermodynamic properties of homogeneous fluids. Therefore, systems investigated with *ms2* typically contain on the order of 10^3 molecules. While for Monte Carlo simulations a perfect scaling behavior up to large numbers of cores can be trivially achieved, MD domain decomposition – the de facto standard for highly scalable MD – is not feasible for such system sizes, because the cut-off radius is in the same range as half the edge length of the simulation volume. This excludes domain decomposition and limits the scalability of the MPI parallelization. The present release of *ms2* features an OpenMP parallelization, which was hybridized with MPI. At the point where MPI communication becomes a bottleneck, a single process still has enough load to distribute to multiple threads, improving scalability.

Three parts of *ms2* were parallelized with OpenMP: the interaction partner search, the energy and the force calculations. All OpenMP parallel regions rely on loop parallelism, as the compute intensive parts of the algorithm all feature a loop over the molecules. In the force calculation, race conditions need to be considered, because every calculated force is written to both interacting molecules. Introducing atomic updates or critical sections leads to massive overheads. Instead, it is more efficient to assign forces from individual interactions to the elements of a list (or an array) which is subsequently summed up. The same holds true for torques.

In Figure 1 the speed-up of hybrid MPI/OpenMP vs. pure MPI is plotted for 2'048 cores, varying the number of threads per MPI process and the number of molecules in the simulation volume. As can be seen, using 2 to 4 threads per MPI process delivers a speed-up of around 20% for 2'048 cores. The evaluation of the hybrid parallelization algorithm was performed on a CRAY XE6 Supercomputer at the High Performance Computing Center in Stuttgart, which has an overall peak performance of one PFLOPS. It consists of 3552 nodes, each equipped with two AMD Opteron 6276 (Interlagos) processors. Each processor has 16 cores, sharing eight FPUs (Floating Point Units). Nodes are equipped with 32 GB RAM and are interconnected by a high-speed CRAY Gemini network. Additional runtime performance comparisons with the simulation tool *gromacs* [3] are listed in Table 1.

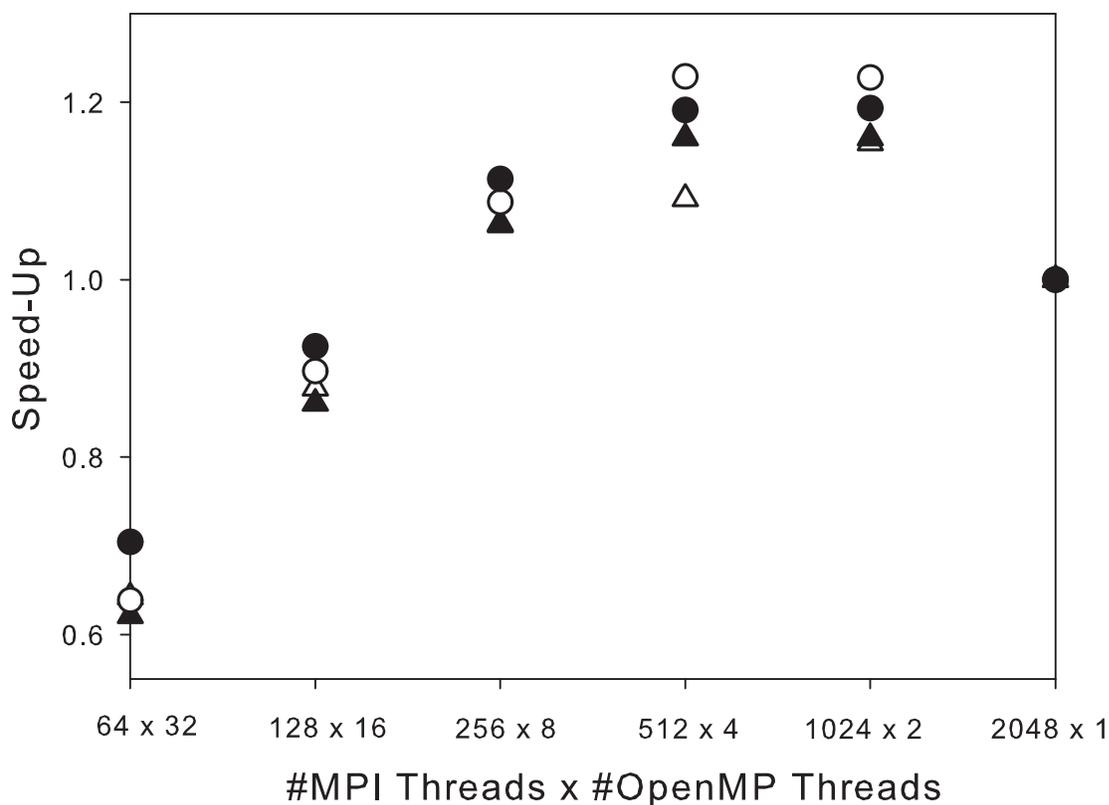


Figure 1: Speed-up of hybrid MPI/OpenMP vs. pure MPI for 2048 cores, varying number of threads per MPI process and 8192 molecules (solid circles), 4096 molecules (empty circles), 2048 molecules (solid triangles), 1024 molecules (empty triangles)

Table 1: Runtime performance results with *ms2* release 2.0 and gromacs v4.6.5 [3] for MD simulations with pure water at 298.15 K and $55.345 \text{ mol} \cdot \text{dm}^{-3}$. The number of time steps were 100 000 for every simulation, the cutoff radius was identical for simulations with the same number of particles. All simulations were performed on the same computer cluster.

cores	threads	N	gromacs / s	<i>ms2</i> (RF) / s	<i>ms2</i> (EW) / s
8	8 MPI	500	164	416	785
8	8 MPI	1000	299	874	1607
8	8 MPI	2000	1284	4461	6777
16	16 MPI	500	95	233	415
16	16 MPI	1000	166	477	848
16	16 MPI	2000	678	2298	3506
32	32 MPI	500	62	152	245
32	32 MPI	1000	106	296	487
32	32 MPI	2000	361	1286	1898
64	64 MPI	500	40	119	166
64	64 MPI	1000	65	228	324
64	64 MPI	2000	220	814	1261
128	128 MPI	500	38	105	131
128	128 MPI	1000	51	197	247
128	128 MPI	2000	147	557	727
8	1 MPI, 8 OMP/MPI	500	167	483	
8	1 MPI, 8 OMP / MPI	1000	323	975	
8	1 MPI, 8 OMP / MPI	2000	1416	4831	
16	2 MPI, 8 OMP / MPI	500	105	253	
16	2 MPI, 8 OMP / MPI	1000	186	517	
16	2 MPI, 8 OMP / MPI	2000	763	2514	
32	4 MPI, 8 OMP / MPI	500	75	167	
32	4 MPI, 8 OMP / MPI	1000	121	316	
32	4 MPI, 8 OMP / MPI	2000	418	1362	
64	8 MPI, 8 OMP / MPI	500	60	119	
64	8 MPI, 8 OMP / MPI	1000	92	217	
64	8 MPI, 8 OMP / MPI	2000	261	785	
128	16 MPI, 8 OMP / MPI	500	49	101	
128	16 MPI, 8 OMP / MPI	1000	74	172	
128	16 MPI, 8 OMP / MPI	2000	170	496	

(N) Number of water molecules

(RF) simulations with reaction field correction.

(EW) simulations with Ewald summation.

4. Massieu potential derivatives

ms2 version 2.0 features evaluating free energy derivatives in a systematic manner, thus greatly extending the thermodynamic property types that can be sampled in single simulation runs. The approach is based on the fact that the fundamental equation of state contains the complete thermodynamic information about a system, which can be expressed in terms of various thermodynamic potentials [4], e.g. internal energy $E(N, V, S)$, enthalpy $H(N, p, S)$, Helmholtz free energy $F(N, V, T)$ or Gibbs free energy $G(N, p, T)$, with number of particles N , volume V , pressure p , temperature T and entropy S . These representations are equivalent in the sense that any other thermodynamic property is essentially a combination of derivatives of the chosen form with respect to its independent variables. The form $F/T(N, V, 1/T)$, known as the Massieu potential, is preferred in molecular simulations due to practical reasons [5, 6]. The statistical mechanical formalism of Lustig allows for the simultaneous sampling of any A_{mn}^r in a single NVT ensemble simulation for a given state point [5, 6, 7, 8]

$$\frac{\partial^{m+n}(F/(RT))}{\partial \beta^m \partial \rho^n} \beta^m \rho^n \equiv A_{mn} = A_{mn}^i + A_{mn}^r, \quad (1)$$

where R is the gas constant, $\beta \equiv 1/T$ and $\rho \equiv N/V$. A_{mn} can be separated into an ideal part A_{mn}^i and a residual part A_{mn}^r [9]. The calculation of the residual part is the target of molecular simulation and the derivatives A_{10}^r , A_{01}^r , A_{20}^r , A_{11}^r , A_{02}^r , A_{30}^r , A_{21}^r and A_{12}^r were implemented in *ms2* for NVT ensemble simulations. The ideal part can be obtained by independent methods, e.g. from spectroscopic data or ab initio calculations. However, it can be shown that for any $A_{mn} = A_{mn}^i + A_{mn}^r$, where $n > 0$, the ideal part is either zero or depends exclusively on the density, thus it is known by default [6]. Note that the calculation of A_{00}^r still requires additional concepts

such as thermodynamic integration or particle insertion methods. From the first five derivatives A_{10} , A_{01} , A_{20} , A_{11} , A_{02} every measurable thermodynamic property can be expressed (see the supplementary material for a list of properties) with the exception of phase equilibria. A detailed description of the implementation is in the supplementary material, here, only an overview is provided.

The calculation of the derivatives up to the order of $n = 2$ requires the explicit mathematical expression of $\partial U/\partial V$ and $\partial^2 U/\partial V^2$ with respect to the applied molecular interaction pair potential and has to be determined analytically beforehand [5, 6]. The general formula for $\partial^n U/\partial V^n$ can be found in Ref. [8]. For common molecular interaction pair potentials, like the Lennard-Jones potential [10, 11], describing repulsive and dispersive interactions, or Coulomb’s law, describing electrostatic interactions between point charges, the analytical formulas for $\partial U/\partial V$ and $\partial^2 U/\partial V^2$ can be obtained straightforwardly.

As molecular simulation is currently limited to operate with considerably fewer particles than real systems, the effect of the small system size thus has to be counter-balanced with a contribution to U and $\partial^n U/\partial V^n$ called long range correction (LRC) [10, 11]. The mathematical form of the LRC depends on the molecular interaction potential and the cut-off method (site-site or center-of-mass cut-off mode) applied. For the Lennard-Jones potential, the LRC scheme was well described in the literature for both the site-site [5, 12] and the center of mass cut-off mode [8, 13]. The reaction field method [14] was the default choice in the preceding version of *ms2* for the LRC of electrostatic interactions modelled by considering charge distributions on molecules. The usual implementation of the reaction field method combines the explicit and the LRC part in a single pair potential [14, 15] from which $\partial^n U/\partial V^n$ (including the LRC contribution) is directly obtainable. However, practical applications show that the electrostatic LRC of $\partial U/\partial V$ and $\partial^2 U/\partial V^2$ can be neglected in case of systems for which the reaction field method is an appropriate choice. E.g., the contribution of the electrostatic LRC for a liquid system ($T = 298$ K and $\rho = 45.86$ mol/l) consisting of only 200 water and 50 methanol molecules with a very short cut-off radius of 20% of the edge length of the simulation volume is still $\ll 1\%$ for both $\partial U/\partial V$ and $\partial^2 U/\partial V^2$. The supplementary material contains detailed elaborations on the LRC for the Lennard-Jones potential.

5. Algorithmic Developments

Transport property calculations. In *ms2*, transport properties are determined via equilibrium MD simulations by means of the Green-Kubo formalism [16]. This formalism offers a direct relationship between transport coefficients and the time integral of the autocorrelation function of the corresponding fluxes. An extended time step was defined for the calculation of the fluxes, the autocorrelation functions and their integrals. The extended time step is n times longer than the specified MD time step, where n is a user defined variable. The autocorrelation functions are hence evaluated in every n -th MD time step. As a consequence, the memory demand for the autocorrelation functions was reduced and the restart files, which contain the current state of the autocorrelation functions and time integrals, become accordingly smaller. In addition, the overall computing time of the MD simulation was reduced significantly.

Ewald summation. Ewald summation [10, 11] was implemented for the calculation of electrostatic interactions between point charges. It extends the applicability of *ms2* to thermodynamic properties of e.g. ions in solutions. In Ewald summation, the electrostatic interactions according to Coulomb’s law are divided into two contributions: short-range and long-range. The short-range term includes all charge-charge interactions at distances smaller than the cut-off radius. The remaining contribution is calculated in Fourier space and only the final value is transformed back into real space. This allows for an efficient calculation of the long-range interactions between the charges. The algorithm is well described in literature. Currently, some of the new features, the calculation of Massieu potential derivatives and Hybrid MPI & OpenMP Parallelization for MD, are not available together with Ewald summation.

6. Property Calculations

Radial distribution function. The radial distribution function (RDF) $g(r)$ is a measure for the microscopic structure of matter. It is defined by the local number density around a given position within a molecule $\rho^L(r)$ in relation to the overall number density $\rho = N/V$

$$g(r) = \frac{\rho^L(r)}{\rho} = \frac{1}{\rho} \frac{dN(r)}{dV} = \frac{1}{4\pi r^2 \rho} \frac{dN(r)}{dr}. \quad (2)$$

Therein, $dN(r)$ is the differential number of molecules in a spherical shell volume element dV , which has the width dr and is located at the distance r from the regarded position. $g(r)$ can be evaluated for every molecule of a given species.

In the present release of *ms2*, the RDF can be calculated during MD simulation runs for pure components and mixtures on the fly. The RDF is sampled between all LJ sites. In order to evaluate RDFs for arbitrary positions, say point charge sites, superimposed dummy LJ sites with the parameters $\sigma = \epsilon = 0$ have to be introduced in the potential model file by the user.

Electric conductivity. The evaluation of the electric conductivity σ was implemented in *ms2* version 2.0, being a measure for the flow of ions in solution. The Green-Kubo formalism [16] offers a direct relationship between σ and the time-autocorrelation function of the electric current flux $\mathbf{j}_e(t)$ [17]

$$\sigma = \frac{1}{3Vk_B T} \int_0^\infty \langle \mathbf{j}_e(t) \cdot \mathbf{j}_e(0) \rangle dt, \quad (3)$$

where k_B is Boltzmanns constant. The electric current flux is defined by the charge q_k of ion k and its velocity vector \mathbf{v}_k according to

$$\mathbf{j}_e(t) = \sum_{k=1}^{N_j} q_k \cdot \mathbf{v}_k(t), \quad (4)$$

where N_j is the number of molecules of component j in solution. Note that only the ions in the solution have to be considered, not the electro-neutral molecules. For better statistics, σ is sampled over all independent spatial elements of $\mathbf{j}_e(t)$.

Thermal conductivity of mixtures. In the previous version of *ms2* the determination of the thermal conductivity by means of the Green-Kubo formalism was implemented for pure substances only. In the present release, the calculation of the thermal conductivity was extended to multi-component mixtures. The thermal conductivity λ is given by the autocorrelation function of the elements of the microscopic heat flow J_q^x

$$\lambda = \frac{1}{Vk_B T^2} \int_0^\infty dt \langle J_q^x(t) \cdot J_q^x(0) \rangle. \quad (5)$$

In mixtures, energy transport and diffusion occur in a coupled manner, thus, the heat flow for a mixture of n components is given by [18]

$$\begin{aligned} \mathbf{J}_q &= \frac{1}{2} \sum_{i=1}^n \sum_{k=1}^{N_i} \left[m_i^k (v_i^k)^2 + \mathbf{w}_i^k \mathbf{\Gamma}_i^k \mathbf{w}_i^k + \sum_{j=1}^n \sum_{l \neq k}^{N_j} u(r_{ij}^{kl}) \right] \cdot \mathbf{v}_i^k \\ &- \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^{N_i} \sum_{l \neq k}^{N_j} \mathbf{r}_{ij}^{kl} \cdot \left(\mathbf{v}_i^k \cdot \frac{\partial u(r_{ij}^{kl})}{\partial \mathbf{r}_{ij}^{kl}} + \mathbf{w}_i^k \mathbf{\Gamma}_i^{kl} \right) - \sum_{i=1}^n h_i \sum_{k=1}^{N_i} \mathbf{v}_i^k, \end{aligned} \quad (6)$$

where \mathbf{w}_i^k is the angular velocity vector of molecule k of component i and $\mathbf{\Gamma}_i^k$ its matrix of angular momentum of inertia. $u(r_{ij}^{kl})$ is the intermolecular potential energy and $\mathbf{\Gamma}_i^{kl}$ is the torque due to the interaction of molecules k and l . The indices i and j denote the components of the mixture. h_i is the partial molar enthalpy. It has to be specified as an input in the *ms2* parameter file and can be calculated from NpT simulations.

Residence time. The residence time τ_j defines the average time span that a molecule of component j remains within a given distance r_{ij} around a specific molecule i . It is given by the autocorrelation function

$$\tau_j = \int_{t=0}^\infty \left\langle \frac{1}{n_{ij}(0)} \sum_{k=1}^{n_{ij}(0)} \Theta_k(t) \Theta_k(0) \right\rangle dt, \quad (7)$$

where t is the time, $n_{ij}(0)$ the solvation number around molecule i at $t = 0$ and Θ is the Heaviside function, which yields unity, if the two molecules are within the given distance, and zero otherwise. Following the proposal of Impey et al. [19], the residence time explicitly allows for short time periods during which the distance between the two molecules exceeds r_{ij} . Also, the solvation number n_{ij} can be evaluated on the fly

$$n_{ij} = 4\pi\rho_j \int_0^{r_{\min}} r^2 g_{ij}(r) dr, \quad (8)$$

where ρ_j is the number density of component j and r_{\min} is the distance up to which the solvation number is calculated.

Acknowledgments

The authors gratefully acknowledge financial support by the BMBF "01IH13005A SkaSim: Skalierbare HPC-Software für molekulare Simulationen in der chemischen Industrie" and computational support by the High Performance Computing Center Stuttgart (HLRS) under the grant MMHBF2. The present research was conducted under the auspices of the Boltzmann-Zuse Society of Computational Molecular Engineering (BZS).

- [1] S. Deublein, B. Eckl, J. Stoll, S. V. Lishchuk, G. Guevara-Carrion, C. W. Glass, T. Merker, M. Bernreuther, H. Hasse, J. Vrabec, *Comp. Phys. Comm.* 182 (2011) 2350–2367.
- [2] G. Rutkai, M. Thol, R. Lustig, R. Span, J. Vrabec, *J. Chem. Phys.* 139 (2013) 041102.
- [3] Gromacs molecular dynamics simulation package, <http://www.gromacs.org>.
- [4] A. Münster, *Classical Thermodynamics*, Wiley and Sons, Bristol, 1970.
- [5] R. Lustig, *Mol. Sim.* 37 (2011) 457–465.
- [6] R. Lustig, *Mol. Phys.* 110 (2012) 3041–3052.
- [7] R. Lustig, *J. Chem. Phys.* 100 (1994) 3048–3059.
- [8] R. Lustig, *J. Chem. Phys.* 100 (1994) 3060–3067.
- [9] J. S. Rowlinson, F. L. Swinton, *Liquids and Liquid Mixtures*, 3rd ed., Butterworths, London, 1982.
- [10] M. Allen, D. Tildesley, *Computer Simulation of Liquids*, Clarendon Press, Oxford, 1987.
- [11] D. Frenkel, B. Smith, *Understanding Molecular Simulation*, Academic Press, San Diego, 1993.
- [12] K. Meier, S. Kabelac, *J. Chem. Phys.* 124 (2006) 064104.
- [13] R. Lustig, *Mol. Phys.* 65 (1998) 175–179.
- [14] J. A. Barker, R. O. Watts, *Mol. Phys.* 26 (1973) 789–792.
- [15] I. P. Omelyan, *Phys. Lett. A* 223 (1996) 295–302.
- [16] K. Gubbins, *Statistical Mechanics*, Burlington House, London, 1972.
- [17] J. P. Hansen, I. R. McDonald, *Theory of Simple Liquids*, Academic, New York, 1986.
- [18] D. J. Evans, W. B. Streett, *Mol. Phys.* 36 (1978) 161–176.
- [19] R. Impey, M. Madden, I. McDonald, *J. Phys. Chem.* 87 (1983) 5071.