

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, B. Eckl, J. Stoll, S. V. Lishchuk, G. Guevara-Carrion, C. W. Glass, T. Merker, M. Bernreuther, H. Hasse, J. Vrabec, *Comput. Phys. Commun.* 182 (2011) 2350] is presented. Version 2.0 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 free energy 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 on electro-neutral molecules, Ewald summation is implemented to consider the long range interactions. Finally, the sampling of the radial distribution function is added.

Keywords: Molecular simulation, molecular dynamics, Monte-Carlo, grand equilibrium method, vapor-liquid equilibrium, transport properties, free energy 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, Windows

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, grand equilibrium method, vapor-liquid equilibrium, transport properties, parallel algorithms

Programming language used: Fortran90

External: Message passing interface (MPI).

Classification: 7.7, 7.9, 12

Parallelized: Yes. Message Passing Interface (MPI) protocol and OpenMP.

Scalability: Scalability up to 2'000 cores for molecular dynamics, depending on the simulation scenario, and more for Monte-Carlo simulations.

Nature of problem: Calculation of application oriented thermodynamic properties for 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.

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 on the basis of experimental data. In the previous release [1], we have introduced the molecular simulation tool *ms2*. Results from *ms2* have been thoroughly 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 scalability. Furthermore, the new release offers a wider scope of accessible properties. In particular, *ms2* was extended to calculate free energy derivatives in a systematic manner, cf. section 4. This augments the range of obtainable data significantly and, as was demonstrated in [2], it allows to 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 has been reduced significantly (section 5).

ms2 is freely available as 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 of 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 new release of *ms2* features an OpenMP parallelization, which is 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. The most efficient way proved to be updating those molecules directly, over which the loop iterates, reducing forces on their interaction partners onto temporary force vectors and summing up after the entire force calculations are done. 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 2-4

threads per MPI process delivers a speed-up of around 20% for 2'048 cores.

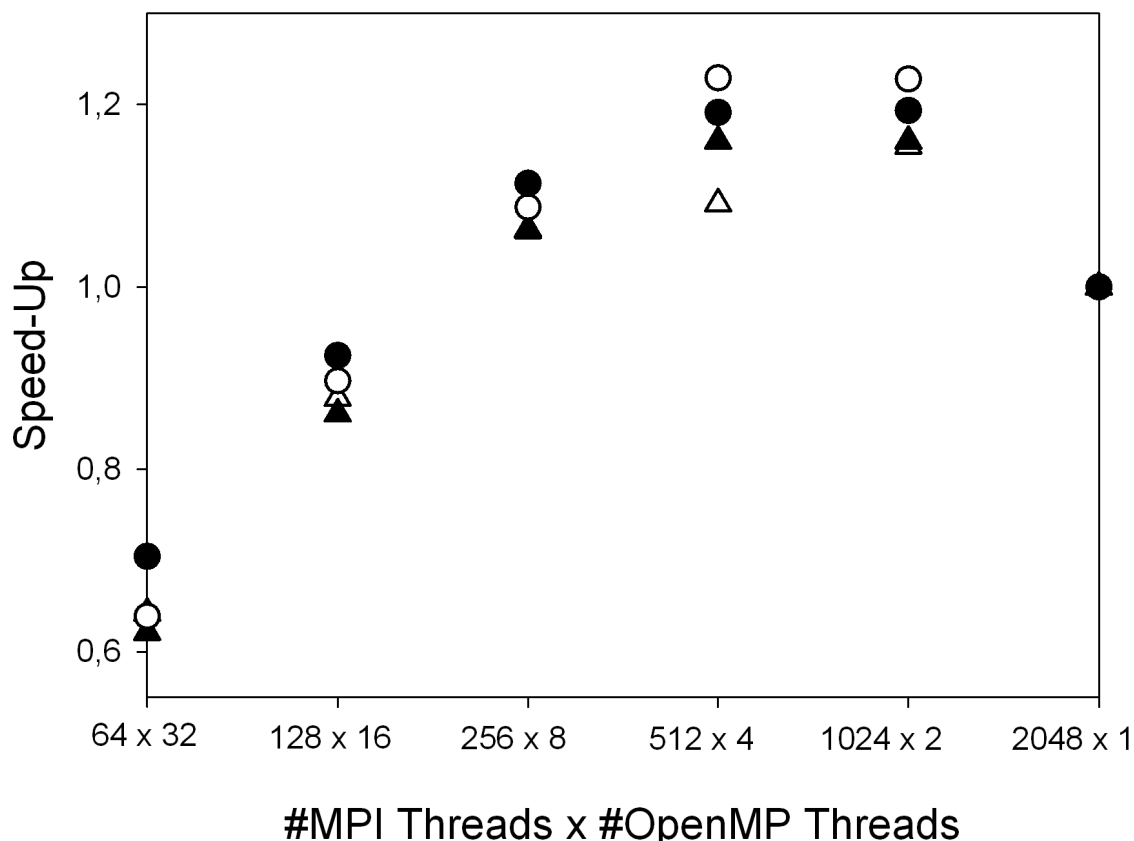


Figure 1: Speed-up of hybrid MPI/OpenMP vs. pure MPI for 2'048 cores, varying number of threads per MPI process and 8'192 Mol. (solid circle), 4'096 Mol. (empty circle), 2'048 Mol. (solid triangle), 1'024 Mol. (empty triangle)

4. Free Energy Derivatives

ms2 Version 2.0 features evaluating free energy derivatives in a systematic manner, thus greatly extending the thermodynamic data that can be sampled from one simulation run. The approach is based on the fact that the fundamental equation of state contains the complete thermodynamic property information about a system and it can be expressed with various thermodynamic potentials [3], e.g. internal energy $E(N, V, S)$, enthalpy $H(N, p, S)$, Helmholtz energy $F(N, V, T)$ or Gibbs 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 function, is preferred in molecular simulations due to practical purposes[4, 5]. 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 [4, 5, 6, 7], where

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

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 [8]. 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 a constant number defined by the density of the system, thus it is known by default [5]. Note that the calculation of A_{00}^r still requires additional concepts such as thermodynamic integration or particle insertion methods and it is not generalized. From the first five derivatives A_{10} , A_{01} , A_{20} , A_{11} , A_{02} basically every currently measurable thermodynamic property can be expressed (see the supplementary material for the list of properties). A detailed description of the implementation is given in the supplementary material, here, only an overview is given.

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 [4, 5]. The general formula for $\partial^n U/\partial V^n$ can be found in Ref. [7]. For common molecular interaction pair potentials, like the Lennard-Jones potential [9, 10], 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 is counter-balanced with a contribution to U and $\partial^n U/\partial V^n$ called long range correction (LRC) [9, 10]. The mathematical form of the LRC depends on the applied molecular interaction potential as well as on the cut-off method (site-site or center of mass cut-off mode). For the Lennard-Jones potential, the LRC scheme was well described in the literature for both the site-site [4, 11] and the center of mass cut-off mode [7, 12]. The reaction field method [13] was the default choice in the preceding version 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 [13, 14] from which $\partial^n U/\partial V^n$ (including also 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 for 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 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 also contains detailed elaborations on the LRC.

5. Algorithmic Developments

Transport Property Calculations. In *ms2*, transport properties are determined via equilibrium MD simulations by means of the Green-Kubo formalism [15]. This formalism offers a direct relationship between transport coefficient and the time integral of the autocorrelation function of the corresponding fluxes. In the present release, these fluxes may be evaluated every n -th time step of the MD simulation. Hence an extended time step is defined for the calculation of the autocorrelation functions and their integrals, which is n times longer than the MD time step. As a consequence, the memory demand of the autocorrelation functions and their time integrals is reduced by a factor of n and the file size of the restart files, which contain the actual state of the autocorrelation

functions and time integrals at the end of the simulation, become accordingly smaller. In addition, the overall computing time of the MD simulation is reduced significantly.

The number n is user-defined: For $n < 6$, no influence of n on the different transport coefficients or the statistical uncertainty was observed.

Ewald Summation. Ewald summation 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 r_c . This contribution is determined explicitly assuming pairwise additivity of the interactions. 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. Eq. (2) defines the total charge-charge interactions

$$\begin{aligned}
u^{\text{coul}} = & \underbrace{\frac{1}{4\pi\epsilon_0} \sum_n' \sum_i \sum_j \sum_{l=1}^{N_{C,i}} \sum_{m=l+1}^{N_{C,j}} q_l q_m \frac{\text{erfc}(\kappa|r_{lm} + \mathbf{n}|)}{|r_{lm} + \mathbf{n}|}}_{\text{Short-range contribution}} \\
& + \underbrace{\frac{1}{\epsilon_0 V} \sum_{k>0} \frac{1}{k^2} e^{-\frac{k^2}{4\kappa^2}} \left(\left| \sum_i \sum_{l=1}^{N_{C,i}} q_l \cos(\mathbf{k} \cdot r_l) \right|^2 + \left| \sum_j \sum_{m=1}^{N_{C,j}} q_m \sin(\mathbf{k} \cdot r_m) \right|^2 \right)}_{\text{Reciprocal term}} \\
& - \underbrace{\frac{1}{4\pi\epsilon_0} \sum_{i=1}^{N_C} \sum_{l=1}^{N_C} \sum_{m=l+1}^{N_C} q_l q_m \frac{\text{erf}(\kappa|r_{lm}|)}{|r_{lm}|}}_{\text{Intramolecular self-energy}} - \underbrace{\frac{\kappa}{4\pi^{1.5}\epsilon_0} \sum_i \sum_{l=1}^{N_C} q_l^2}_{\text{Point self-energy}}, \tag{2}
\end{aligned}$$

where κ is the real-space partition parameter, r_{lm} the distance between the two charges q_l and q_m , \mathbf{k} the reciprocal lattice vector and k its modulus.

The algorithm is well described in literature [9, 16].

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 $\rho = N/V$

$$g(r) = \frac{\rho^L(r)}{\rho} = \frac{1}{\rho} \frac{dN(r)}{dV} = \frac{1}{4\pi r^2 \rho} \frac{dN(r)}{dr}. \tag{3}$$

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 new release of *ms2*, RDF can be calculated during MD simulation runs for pure components and mixtures on the fly. It is sampled between all LJ sites, which are defined by the potential model *.pm files. In order to evaluate RDFs for arbitrary positions, dummy LJ sites with the parameters $\sigma = \epsilon = 0$ can be introduced.

Electric conductivity. The evaluation of the electric conductivity σ was implemented in *ms2*, being a measure for the flow of ions in solution. The Green-Kubo formalism [15] offers a direct relationship between σ and the time-autocorrelation function of the flux of the electric current $\mathbf{j}(t)$

$$\sigma = \frac{1}{3Vk_{\text{B}}T} \int_0^{\infty} \langle \mathbf{j}(t) \cdot \mathbf{j}(0) \rangle dt, \quad (4)$$

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

$$\mathbf{j}(t) = \sum_{k=1}^{N_j} q_k \cdot \mathbf{v}_k(t), \quad (5)$$

where N_j is the number of molecules of component j in solution. Note that all ions in the solution have to be considered, but not the electro-neutral molecules. For better statistics, σ is determined over all independent spatial elements of $\mathbf{j}(t)$. The electric current time-autocorrelation function may be decomposed into the sum [17]

$$\begin{aligned} \langle \mathbf{j}(t) \cdot \mathbf{j}(0) \rangle &= \sum_{k=1}^{N_{\text{Ion}}} \langle q_k^2 \cdot \mathbf{v}_k(t) \cdot \mathbf{v}_k(0) \rangle + \sum_{k=1}^{N_{\text{Ion}}} \sum_{\substack{n=1 \\ n \neq k}}^{N_{\text{Ion}}} \langle q_k q_n \cdot \mathbf{v}_k(t) \cdot \mathbf{v}_n(0) \rangle \\ &= Z(t) + \Delta(t), \end{aligned} \quad (6)$$

where $Z(t)$ is an autocorrelation function and $\Delta(t)$ is a crosscorrelation that quantifies the deviation from the ideal Nernst-Einstein behavior [17].

The first term $Z(t)$ describes the mobility of the ions due to their self-diffusion in solution. Mathematically, it is simply the sum of the self-diffusion coefficients of all ion types in solution weighted by their charges. The second term $\Delta(t)$ describes the correlated motion of the ions in solution. E.g., the correlated motion of ion pairs of opposite charges in solution lowers the electric conductivity ($\Delta(t) < 0$), while the correlated motion of ion pairs with the same charge enlarge σ ($\Delta(t) > 0$).

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

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

where t is the time and Θ is the Heaviside function, which yields unity, if the two molecules are within the given distance, and zero if not. Following the proposal of Impey et al. [18], the residence time explicitly allows for short time periods, during which the distance between the two molecules exceeds r_{ij} . Also, the hydration 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 hydration 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 MMHBF. The present research was conducted under the auspices of the Boltzmann-Zuse Society of Computational Molecular Engineering (BZS).

Bibliography

- [1] Deublein, S. et al., *Comp. Phys. Comm.* **182** (2011) 2350.
- [2] Rutkai, G., Thol, M., Lustig, R., Span, R., and Vrabc, J., *J. Chem. Phys.* **139** (2013) 041102.
- [3] Münster, A., *Classical Thermodynamics*, John Wiley and Sons, Bristol, 1970.
- [4] Lustig, R., *Mol. Sim.* **37** (2011) 457.
- [5] Lustig, R., *Mol. Phys.* **110** (2012) 3041.
- [6] Lustig, R., *J. Chem. Phys.* **100** (1994) 3048.
- [7] Lustig, R., *J. Chem. Phys.* **100** (1994) 3060.
- [8] Rowlinson, J. S. and Swinton, F. L., *Liquids and Liquid mixtures, 3rd ed.*, Butterworths, London, 1982.
- [9] Allen, M. and Tildesley, D., *Computer Simulation of Liquids*, Clarendon Press, Oxford, 1987.
- [10] Frenkel, D. and Smit, B., *Understanding Molecular Simulation: From Algorithms to Applications*, Academic Press, Elsevier, San Diego, 2002.
- [11] Meier, K. and Kabelac, S., *J. Chem. Phys.* **124** (2006) 064104.
- [12] Lustig, R., *Mol. Phys.* **65** (1998) 175.
- [13] Barker, J. A. and Watts, R. O., *Mol. Phys.* **26** (1973) 789.
- [14] Omelyan, I. P., *Phys. Lett. A* **223** (1996) 295.
- [15] Gubbins, K., *Statistical Mechanics*, The Chemical Society Burlington House, London, 1972.
- [16] Frenkel, D. and Smith, B., *Understanding Molecular Simulation*, Academic Press, Elsevier, San Diego, 1993.

[17] Del Popolo, M. and Voth, G., J. Phys. Chem. B **108** (2004) 1744.

[18] Impey, R., Madden, M., and McDonald, I., J. Phys. Chem. **87** (1983) 5071.