*ms*2: A molecular simulation tool for thermodynamic properties, release 3.0

Gábor Rutkai[a], Andreas Köster[a], Gabriela Guevara-Carrion[a], Tatjana Janzen[a], Michael Schappals[b], Colin W. Glass[c], Martin Bernreuther[c], Amer Wafai[c], Simon Stephan[b], Maximilian Kohns[b], Steffen Reiser[b], Stephan Deublein[b], Martin Horsch[b], Hans Hasse[b], Jadran Vrabec[a,*]

[a] *Lehrstuhl für Thermodynamik und Energietechnik, Universität Paderborn, 33098 Paderborn, Germany*
[b] *Lehrstuhl für Thermodynamik, Universität Kaiserslautern, 67653 Kaiserslautern, Germany*
[c] *Höchstleistungsrechenzentrum Universität Stuttgart (HLRS), 70550 Stuttgart, Germany*

Abstract

A new version release (3.0) of the molecular simulation tool *ms*2 [Deublein et al., Comput. Phys. Commun. 182 (2011) 2350 and Glass et al., Comput. Phys. Commun. 185 (2014) 3302] is presented. Version 3.0 of *ms*2 features two additional ensembles, i.e. microcanonical (NVE) and isobaric–isoenthalpic (NpH), various Helmholtz energy derivatives in the NVE ensemble, thermodynamic integration as a method for calculating the chemical potential, the six Maxwell-Stefan diffusion coefficients of quaternary mixtures, statistics for sampling hydrogen bonds, the osmotic pressure for calculating the activity of solvents, smooth-particle mesh Ewald summation as well as the ability to carry out molecular dynamics runs for an arbitrary number of state points in a single program execution.

Keywords: molecular simulation, molecular dynamics, Monte Carlo

New version program summary

Program Title: *ms*2

Operating system: Unix/Linux

Licensing provisions: Special license supplied by the authors

Programming language: Fortran95

Has the code been vectorized or parallelized: yes (Message Passing Interface (MPI) protocol and OpenMP)

*Corresponding author: Jadran Vrabec, Warburger Str. 100, 33098 Paderborn, Germany, Tel.: +49-5251/60-2420, Email: jadran.vrabec@upb.de

Distribution format: tar.gz

Classification: 7.7, 7.9, 12

Catalogue identifiers of previous versions: AEJF_v1_0 and AEJF_v2_0

Supplementary material: A detailed description of the parameter setup for thermodynamic integration and hydrogen bonding is given in the supplementary material. Furthermore, all molecular force field models developed by our group are provided

Journal reference of previous versions: Deublein et al., Comput. Phys. Commun. 182 (2011) 2350 and Glass et al., Comput. Phys. Commun. 185 (2014) 3302

Does the new version supersede the previous version?: Yes

Reasons for the new version: Introduction of new features as well as enhancement of computational efficiency

Summary of revisions: Two new ensembles (NVE and NpH), new properties (Helmholtz energy derivatives, chemical potential via thermodynamic integration, Maxwell-Stefan diffusion coefficients of quaternary mixtures, activity coefficients via osmotic pressure), new functionalities (detection and statistics of hydrogen bonding, smooth-particle mesh Ewald summation, ability to carry out molecular dynamics runs for an arbitrary number of state points in a single program execution)

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

Solution method: Molecular dynamics, Monte Carlo, various ensembles, Grand Equilibrium method, Green-Kubo formalism, Lustig formalism, OPAS method, smooth-particle mesh Ewald summation

Typical running time: Typically from a couple of hours up to days, depending on the specific scenario (system size, calculated properties, number of CPU cores used)

Restrictions: Typical problems addressed by $ms2$ are solved by simulating systems containing 1000 to 5000 molecules that are modelled as rigid bodies

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

1. Introduction

Due to the continuous increase in computing power, the range of possible applications of molecular modeling and simulation has become broader over time, proceeding from qualitative basic research in soft matter physics to quantitative applications in chemical engineering. Reaching agreement with the available experimental data, and predicting properties where experimental data are rare or absent, molecular methods transform engineering data science [1, 2]. This progress is driven by massively-parallel high performance computing with scalable codes [3] and by the concurrent execution of large numbers of simulations [4] or of simulations which can be decomposed into a large number of concurrent tasks [5–7].

The program $ms2$ (molecular simulation 2) was designed to compute thermodynamic properties of pure fluids and mixtures by Monte Carlo (MC) and molecular dynamics (MD) simulation. Licences are available without cost for all purposes which concern academic research and teaching [8]. The previous two major releases of $ms2$ [9, 10] facilitate the simulation of vapor-liquid equilibria (VLE) by Grand Equilibrium simulation and the computation of many thermodynamic bulk properties, including linear transport coefficients, for molecular models consisting of Lennard-Jones interaction sites, point charges and point multipoles. It has been shown that such models can reach a high accuracy for a wide variety of thermodynamic properties for many molecular fluids [11–16], leading to an increasing popularity of molecular methods in the engineering sciences [17–21].

Similar molecular simulation programs, which address multiple academic communities, include CHARMM [22], DL_POLY [23], ESPResSo [24], GIBBS [25], GROMACS [26], IMD [27], LAMMPS [28], $ls1$ mardyn [29], NAMD [30], TINKER [31] and Towhee [32]. In comparison with these codes, the aim of $ms2$ is to focus on applications of molecular modeling and simulation in fluid process engineering, both industrial and academic. Hence, a high accuracy, short response times and the suitability for coupling with equations of state [4, 33] and rigorous model optimization approaches [34–36] have been priorities in developing both the code base as well as the toolset which is provided jointly with it. The present work discusses the third major release of $ms2$ and its most important innovations, which are presented in detail in the following sections.

2. Microcanonical and isobaric-isoenthalpic ensembles

An ensemble is the set of all theoretically possible microscopic configurations on the molecular level under specific macroscopic constrains. The microcanonical (NVE) ensemble is the set of all configurations that fulfill the condition of having the same number of particles N, volume V and energy E, whereas the isobaric-isoenthalpic (NpH) ensemble represents a system at constant number of particles N, pressure p and enthalpy H. Molecular dynamics (MD) mimics the time evolution of a mechanical system by numerically solving Newton's equations of motion for all considered molecules. Because of the nature of this solution, the time is discredized and the method yields microscopic configurations at discrete and consecutive time steps. The Monte Carlo (MC) method is the application of statistical mechanics to describe molecular systems. With this approach, microscopic configurations are generated by random numbers that are potentially accepted such that only relevant and physically meaningful configurations are sampled [37]. The generation and acceptance of these configurations is governed by specific probabilities that are ensemble-dependent

Table 1: Comparison of the ensembles implemented in *ms*2 for methyl fluoride [41] in terms of temperature T, density ρ, pressure p and potential energy u. Numbers in parentheses denote statistical uncertainties in the last digit that were estimated with the block averaging method of Flyvbjerg and Petersen [42].

		T	ρ	p	u
		K	mol/l	MPa	kJ/mol
MD	NVT	300	1	2.090(2)	-1048(3)
	NpT	300	1.0015(7)	2.090	-1051(3)
	NVE	300.0(1)	1	2.088(1)	-1048(3)
	NpH	300.0(1)	1.0003(2)	2.090	-1048(2)
MC	NVT	300	1	2.0880(2)	-1051.6(2)
	NpT	300	1.0019(2)	2.0880	-1053.6(3)
	NVE	299.99(1)	1	2.0882(2)	-1051.3(2)
	NpH	300.06(2)	1.0010(1)	2.0880	-1051.7(2)

and described in the literature for essentially every relevant ensemble. For MC simulations, the NVE and NpH ensembles were implemented in *ms*2 release 3.0 according to Refs. [38, 39]. For MD simulations, the pressure is kept constant using Andersen's barostat [40] in case of the NpH ensemble. Because the solution of Newton's equations of motion is approximate, the total energy of the system $E = K + U$, which consists of a kinetic K (exclusively molecular momentum dependent) and potential U (exclusively molecular position dependent) energy contribution, is not rigorously conserved in a standard MD run mainly for numerical reasons. Therefore, the translational and rotational momentum of every molecule is rescaled such that the total kinetic energy K fulfils $K = E - U$, where the total energy E is specified and the potential energy U is calculated from the current microscopic configuration. For a NpH ensemble run, the solution is analogous: Momenta are rescaled such that the current kinetic energy K fulfils $K = H - U - pV$, where H and p are specified, U and V are dependent on the current microscopic configuration. This extends the ensembles available in *ms*2 to five: NVT, NVE, NpT, NpH and μVT, where μ is the chemical potential.

For verification purposes, Table 1 contains numerical results for methyl fluoride modeled by a quadrupolar two-center Lennard–Jones potential [41] at $T = 300$ K and $\rho = 1$ mol/l. Figure 1 shows the running averages of the calculated properties in Table 1 at the same state point.

3. Helmholtz energy derivatives in the microcanonical ensemble

The generalized calculation of the Helmholtz energy derivatives

$$A^{\mathrm{r}}_{nm} = (1/T)^n \rho^m \frac{\partial^{n+m} f^{\mathrm{r}}(T,\rho)/(RT)}{\partial(1/T)^n \partial\rho^m}, \tag{1}$$

with the molar residual Helmholtz energy f^{r}, the temperature T, the density ρ and the molar gas constant R was introduced in *ms*2 also for the NVE ensemble up to the order $n = 3$ and

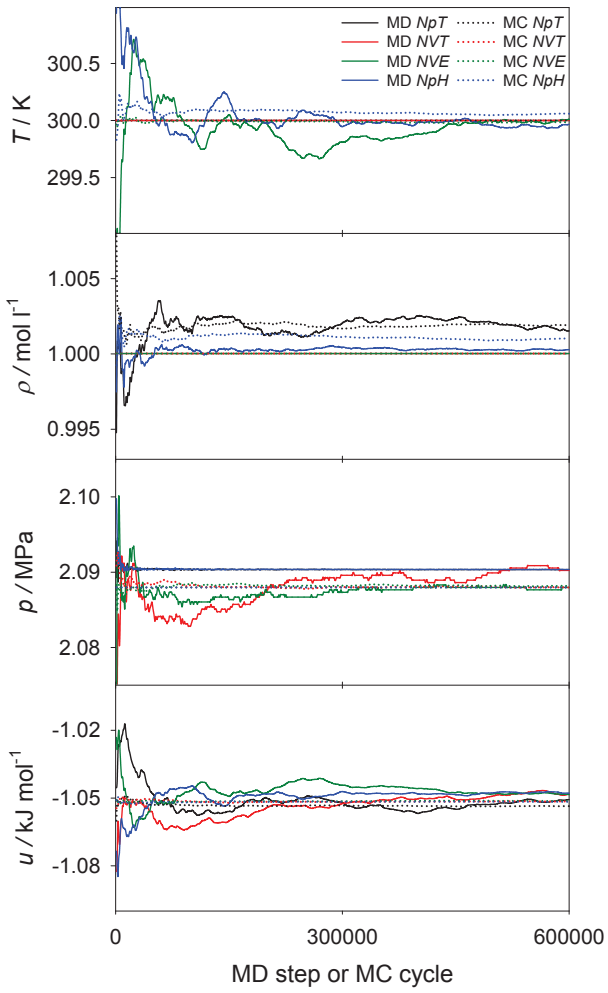

Figure 1: Comparison of the ensembles implemented in *ms*2 for methyl fluoride [41] in terms of running averages for temperature T, density ρ, pressure p and potential energy u.

Table 2: Comparison of the residual Helmholtz energy derivatives A^{r}_{nm} of methyl fluoride [41] at $T = 300$ K and $\rho = 1$ mol/l. Numbers in brackets denote statistical uncertainties in the last digit that were estimated with the block averaging method of Flyvbjerg and Petersen [42]. Superscripts (1) and (2) indicate the two different entropy definitions [45].

		A^{r}_{10}	A^{r}_{01}	A^{r}_{20}	A^{r}_{11}	A^{r}_{02}
	NVT	-0.420(1)	-0.1620(6)	-0.54(4)	-0.40(2)	0.03(3)
MD	NVE^1	-0.418(1)	-0.1622(4)	-0.55(3)	-0.37(1)	0.05(3)
	NVE^2	-0.418(1)	-0.1621(4)	-0.55(3)	-0.37(1)	0.05(3)
	NVT	-0.42158(8)	-0.16290(8)	-0.562(2)	-0.391(2)	0.011(6)
MC	NVE^1	-0.42149(5)	-0.16280(8)	-0.560(2)	-0.385(3)	0.024(6)
	NVE^2	-0.42133(5)	-0.16274(8)	-0.560(2)	-0.385(3)	0.024(6)

$m = 2$. The total reduced Helmholtz energy $f/(RT)$ can be additively separated into an ideal $f^{\mathrm{o}}/(RT)$ and a residual contribution $f^{\mathrm{r}}/(RT)$. The ideal contribution by definition corresponds to the value of $f/(RT)$ when no intermolecular interactions are at work [43]. $f^{\mathrm{o}}/(RT)$ consists of an exclusively temperature and an exclusively density dependent part. The latter is the trivial term $\ln(\rho/\rho_{\mathrm{ref}})$, whereas the former is non-trivial. A formalism that allows for the calculation of the residual Helmholtz derivatives on the fly from a single simulation run per state point was published by Lustig [44, 45] and was already introduced for the NVT ensemble in the preceding program version. $ms2$ release 3.0 now yields these derivatives also for the NVE ensemble. However, in contrast to NVT simulations, there are two numerical results for each calculated derivative A_{nm} due to the two possible entropy definitions in statistical mechanics [45]. In any case, these two different sets of results must be identical in the thermodynamic limit ($N \longrightarrow \infty$). In practice, they already agree within their statistical uncertainty for a simulation based on around a thousand molecules. For verification purposes, Table 2 contains numerical results for methyl fluoride [41]. A detailed description of the calculation of these derivatives can be found in Ref. [45]. Their conversion into common thermodynamic properties is provided in the supplementary material of release 2.0 of $ms2$ [10] and in Ref. [43].

4. Thermodynamic integration

The previously described method of Lustig does not allow for the direct sampling of the chemical potential and other entropic properties like A^{r}_{00}. Such an effort requires techniques based on free energy calculation, such as particle insertion and/or thermodynamic integration [37]. Widom's particle insertion method [46] is a conceptually straightforward approach to calculate the chemical potential with a low computational cost, both for pure substances and mixtures. The total chemical potential μ_i of species i can be separated into an ideal (o) and a residual (r) contribution in the same way as the Helmholtz energy is decomposed, cf. section 2: $\mu_i(T, \rho, x_i) = \mu^{\mathrm{o}}_i(T) + RT \ln(N_i/(V\rho_{\mathrm{ref}})) + \mu^{\mathrm{r}}_i(T, \rho, x_i)$, where N_i is the number of molecules of species i, $\rho = N/V$, $x_i = N_i/N$ and ρ_{ref} is an arbitrary reference density. The expression $\mu_i - \mu^{\mathrm{o}}_i(T)$ is often referred to as the configurational chemical potential μ^{conf}_i. Widom's method requires the frequent insertion of

an additional ($i = N + 1$) test particle into the simulation volume at a random position with a random orientation. At constant temperature and constant pressure or volume the potential energy U_i of this test particle, i.e. the interaction energy with all other "real" N molecules, yields the configurational chemical potential according to

$$\mu_i^{\mathrm{conf}} = \mu_i - \mu_i^{\mathrm{o}}(T) = -k_{\mathrm{B}}T \ln \frac{\langle V \exp\left(-U_i/k_{\mathrm{B}}T\right)\rangle}{\langle N_i \rangle}, \tag{2}$$

where k_{B} is Boltzmann's constant. The test particle is removed immediately after the calculation of its potential energy U_i, thus it does not influence the real molecules in the system in any way. In contrast to the usual convention, the brackets <> have a dual meaning here: They stand for either NVT or NpT ensemble averages as well as an integral over all possible positions and orientations of the test particles added to the system. The density of the system has a significant influence on the accuracy of this method. For state points with a very high density, test particles almost always overlap with real molecules, which leads to a potential energy $U_i \to \infty$ and thus to a vanishing contribution to Eq. (2), resulting in poor statistics and often even to complete failure of sampling.

Thermodynamic integration is one solution to overcome the limitations of Widom's particle insertion method. The idea behind calculating the chemical potential by thermodynamic integration is to avoid insertion of test particles in a challenging system A, but rather perform it in system B for which this can be done without sampling problems. Because the chemical potential is a state property, its difference $\mu_{A,i} - \mu_{B,i}$ can be calculated along any path between states A and B, which is represented by the scalar parameter λ. It can be shown [37] that the relation between λ and $\mu_{A,i} - \mu_{B,i}$ is

$$\mu_{A,i} - \mu_{B,i} = \int_{B(\lambda_{\min})}^{A(\lambda_{\max})} \left\langle \frac{\partial U_i(\lambda)}{\partial \lambda} \right\rangle d\lambda. \tag{3}$$

The brackets <> in this equation denote NVT or NpT ensemble averages and U_i is the potential energy of particle i that must be a part of the system in the same way as the other molecules are. The only difference between particle i and all other molecules is that its interaction energy $U_i(\lambda)$ is scaled between states $A(\lambda_{\max})$ and $B(\lambda_{\min})$ with λ. As long as $\partial U_i(\lambda)/\partial\lambda$ can be calculated analytically and sampled during simulation, the actual way of scaling $U_i(\lambda)$ with λ can be chosen arbitrarily because μ_i is a state property. The integration with respect to λ is carried out numerically. Assuming that $\mu_{B,i}$ is practically zero or at least can successfully be calculated by Widom's particle insertion method for state B using Eq. (2), $\mu_{A,i}$ yields the configurational contribution to the total chemical potential. The non-linear scaling $U_i(\lambda) = \lambda^d U_i$ for $\lambda_{\min} \leq \lambda \leq 1 = \lambda_{\max}$ with an adaptive sampling technique [47] was implemented for MC simulations with d and $\lambda_{\min}$ being input parameters. This adaptive technique allows for the sampling of the entire range of $\lambda_{\min} \leq \lambda \leq 1$ in a single MC run with an arbitrary resolution for numerical integration. In addition to the standard NVT and NpT ensemble MC moves, the simulation includes changes of λ controlled by a proper MC acceptance criterion, ensuring visits at each discrete λ value in the range $\lambda_{\min} \leq \lambda \leq 1$. For MD simulations, changes of λ are also carried out on the fly in a single simulation, but without any acceptance criterion. A detailed description of the parameter setup is given in the supplementary material.

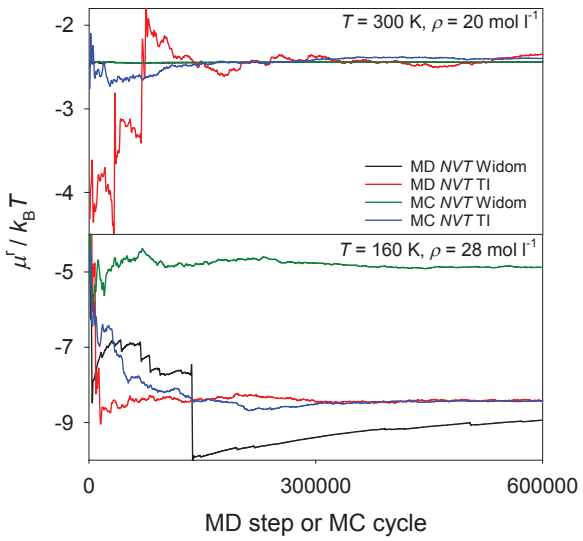

Figure 2: Comparison of the residual chemical potential μ^{r} of methyl fluoride [41] at two different state points calculated by $ms2$ using either Widom's particle insertion or thermodynamic integration (TI).

Figure 2 shows the running averages for the residual chemical potential of methyl fluoride at two different state points calculated by either Widom's particle insertion or thermodynamic integration. For high temperatures and low densities, the agreement between Widom's particle insertion and thermodynamic integration is satisfactory. Widom fails, however, for dense and strongly interacting fluids.

5. Quaternary Maxwell-Stefan diffusion coefficients

$ms2$ employs the Green-Kubo formalism based on the net velocity correlation function to obtain $n \times n$ phenomenological coefficients [48]

$$L_{ij} = \frac{1}{3N} \int_0^\infty \left\langle \sum_{k=1}^{N_i} \mathbf{v}_{i,k}(0) \cdot \sum_{l=1}^{N_j} \mathbf{v}_{j,l}(t) \right\rangle dt, \tag{4}$$

in a mixture of n components. Here, N is the total number of molecules, N_i is the number of molecules of species i and $\mathbf{v}_{i,k}(t)$ denotes the center of mass velocity vector of the k-th molecule of species i at time t. Note that the phenomenological coefficients given in Eq. (4) are constrained by [48]

$$\sum_i M_i L_{ij} = 0\,, \tag{5}$$

where M_i is the molar mass of component i.

Starting from the phenomenological coefficients L_{ij}, the elements of a $(n-1) \times (n-1)$ matrix $\mathbf{\Delta}$ can be defined as [48]

$$\Delta_{ij} = (1 - x_i)\left(\frac{L_{ij}}{x_j} - \frac{L_{in}}{x_n}\right) - x_i \sum_{k=1\neq i}^{n}\left(\frac{L_{kj}}{x_j} - \frac{L_{kn}}{x_n}\right), \tag{6}$$

so that its inverse matrix $\boldsymbol{B} = \boldsymbol{\Delta}^{-1}$ is related to the Maxwell-Stefan diffusion coefficients $Đ_{ij}$. In the case of a quaternary mixture, the six Maxwell-Stefan diffusion coefficients are then given by [48]

$$Đ_{14} = \frac{1}{B_{11} + (x_2/x_1)B_{12} + (x_3/x_1)B_{13}}, \tag{7}$$

$$Đ_{24} = \frac{1}{B_{22} + (x_1/x_2)B_{21} + (x_3/x_2)B_{23}}, \tag{8}$$

$$Đ_{34} = \frac{1}{B_{33} + (x_1/x_3)B_{31} + (x_2/x_3)B_{32}}, \tag{9}$$

$$Đ_{12} = \frac{1}{1/Đ_{24} - B_{21}/x_2}, \tag{10}$$

$$Đ_{13} = \frac{1}{1/Đ_{14} - B_{13}/x_1}, \tag{11}$$

$$Đ_{23} = \frac{1}{1/Đ_{24} - B_{23}/x_2}. \tag{12}$$

$$\tag{13}$$

MD simulation runs for Lennard-Jones fluids were performed in order to test the validity of the quaternary Maxwell-Stefan diffusion coefficients calculated with *ms*2. For this purpose, a quaternary Lennard-Jones pseudo-mixture was created by giving different labels to identical molecules, dividing them into four mole fractions (x_1= 0.1, x_2= 0.2, x_3 = 0.3 and x_4 = 0.4 mol mol^{-1}). A system containing 800 molecules was simulated in the dense liquid state and the resulting Maxwell-Stefan diffusion coefficients were compared with the corresponding self-diffusion coefficient that all have to be identical in the present case.

Fig. 3 shows the development of the calculated values of the six Maxwell-Stefan diffusion coefficients with the number of time origins. As can be seen, the resulting values become independent after around 10^4 time origins and then oscillate around their mean value. The higher statistical uncertainties of $Đ_{12}$ and $Đ_{13}$ are due to the small number of species 1 molecules compared with the number of molecules of the other species.

6. Hydrogen bonding

There is no definite characterization of a hydrogen bond between two molecules [49–51]. Rather, the hydrogen (H) bond 'is a structural motif and involves at least three atoms' [52]. The International Union of Pure and Applied Chemistry (IUPAC) defines the hydrogen bond as 'an attractive interaction between a hydrogen atom from a molecule or molecular fragment X–H in which X is more electronegative than H, and an atom or a group of atoms in the same or a different molecule, in which there is evidence of bond formation' [51].

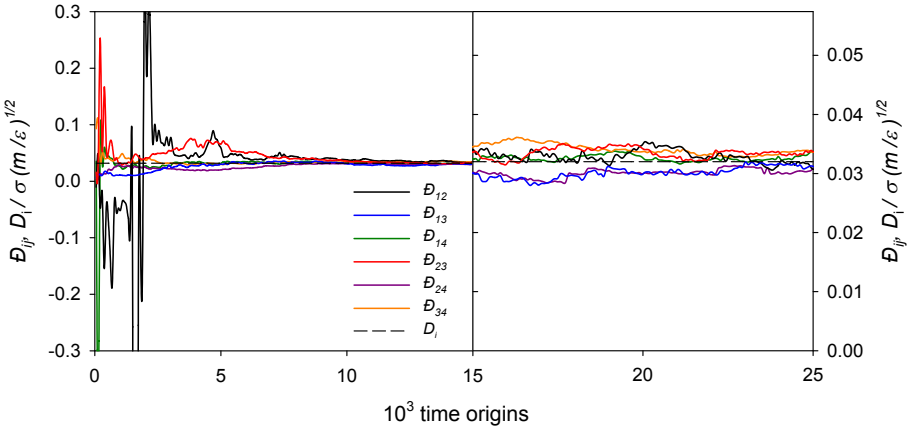

Figure 3: Maxwell-Stefan diffusion coefficients $Đ_{ij}$ and the self-diffusion coefficient D_i as a function of time origins of a quaternary pseudo-mixture ($\epsilon = \epsilon_1 = \epsilon_2 = \epsilon_3 = \epsilon_4$ and $\sigma = \sigma_1 = \sigma_2 = \sigma_3 = \sigma_4$) at $k_\mathrm{B}T/\epsilon = 0.728$ and $\rho\sigma^3 = 0.8442$.

Energetic [49], geometric [53] and topological [50] hydrogen bonding criteria have been proposed in the literature. Geometric criteria, which are not overly complex, are often based on the following assumptions [54]:

- The interaction between two hydrogen bonded sites is highly directional and short ranged.
- A donor interacts at most with a single acceptor, an acceptor may interact with multiple donors.

Accordingly, a class of geometric criteria for the evaluation of hydrogen bonding networks in fluids was implemented in *ms*2. Thereby, the triangle between three charge sites, being part of two different molecules, is evaluated to determine whether two sites constitute a hydrogen bond or not. A molecule acts as a donor to another molecule, i.e. the acceptor, if the following conditions hold [53–56], cf. Fig. 4:

- The distance between the donor and the acceptor is smaller than a threshold distance, i.e. ℓ_AD or ℓ_DA.
- The distance between the acceptor sites of the acceptor and donor molecules is smaller than a threshold distance, i.e. ℓ_AA.
- The angle between the acceptor-donor axis and the acceptor-acceptor axis is smaller than a threshold angle, i.e. φ_DAA or φ_AAD.

Therein, ℓ_AD, ℓ_DA, ℓ_AA, φ_DAA and φ_AAD are parameters of the implemented class of hydrogen bonding criteria. For methanol, e.g., Haughney et al. [53] proposed $\ell_\mathrm{AD} = \ell_\mathrm{DA} = 2.6$ Å, $\ell_\mathrm{AA} = 3.5$ Å and $\varphi_\mathrm{AAD} = \varphi_\mathrm{DAA} = 30°$. Such geometric criteria have been applied to a variety of fluids, in

particular to water [55–57], methanol-water mixtures [58, 59] or ethanol [54, 60, 61]. Hydrogen bonds in sorbate-sorbent interactions can be treated analogously [62]. E.g., for hydrogen fluoride, a simpler distance-based criterion can be used [63], which is also covered by the present implementation. A detailed description of the parameter setup is given in the supplementary material.

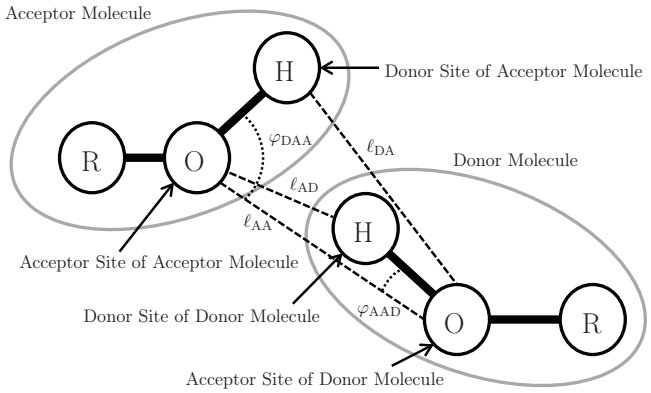

Figure 4: Hydrogen bonding criteria implemented in the present release version of *ms*2.

To test the capabilities of the hydrogen bonding statistics implemented in *ms*2, a MD simulation run of the ternary mixture water (w) + methanol (m) + ethanol (e) ($x_{\mathrm{w}} = 0.33$, $x_{\mathrm{m}} = 0.33$, $x_{\mathrm{e}} = 0.34$ mol mol^{-1}) at $T = 298.15$ K and $p = 0.1$ MPa was carried out with 4000 molecules. In this mixture all species can form hydrogen bonds with each other because all three molecules contain hydroxyl groups. Thus the hydrogen atoms may act as donors and the oxygen atoms as acceptors to form hydrogen bonds. Throughout, between like and unlike molecules, the geometric criteria of Haughney et al. [53] were used. The results are listed in Table 3. The hydrogen bonding statistics in *ms*2 not only indicates the amount of monomers (no bond), dimers (one bond), trimers (two bonds) and tetramers (three bonds), but also provides information about which molecule species are bonded.

7. OPAS Method

The OPAS (osmotic pressure for the activity of the solvent) method was implemented in *ms*2. It is an alternative to e.g. the Widom's particle insertion [46] or thermodynamic integration [64–66], for calculating chemical potentials in the liquid phase by MD simulations. It is particularly well-suited for studying the solvent activity of electrolyte solutions, but can in principle also be used for mixtures of molecular species. Details and a thorough assessment of the method are presented in Ref. [67], the method is only briefly outlined in the following.

The basic idea is a direct simulation of the osmotic equilibrium between a pure solvent phase and a solution phase by introducing semi-permeable membranes into the simulation volume. These are realized by an external force field that acts only on the solute molecules to keep them in the solution phase. By sampling the total net membrane force per membrane area, the pressure difference between the two phases, i.e. the osmotic pressure Π, is sampled. Assuming an incompressible

Table 3: Hydrogen bonding statistics of the ternary mixture water (w) + methanol (m) + ethanol (e) ($x_w = 0.33$, $x_m = 0.33$, $x_e = 0.34$ mol mol^{-1}) at $T = 298.15$ K and $p = 0.1$ MPa in relative terms.

	water	methanol	ethanol
monomer	0.1%	0.9%	1.1%
dimer	1.2%	10.6%	11.5%
bonded to (w)	0.8%	3.5%	4.1%
(m)	0.2%	3.3%	3.6%
(e)	0.2%	3.8%	3.8%
trimer	7.8%	47.7%	42.6%
bonded to (w)(w)	2.8%	2.8%	2.8%
(w)(m)	1.9%	10.5%	8.9%
(w)(e)	2.0%	12.3%	10.2%
(m)(m)	0.3%	5.0%	4.9%
(m)(e)	0.4%	11.0%	10.3%
(e)(e)	0.4%	6.1%	5.5%
tetramer	25.8%	35.9%	36.8%
bonded to (w)(w)(w)	3.9%	15.6%	12.7%
(w)(w)(m)	4.9%	7.6%	7.8%
(w)(w)(e)	5.1%	7.7%	7.6%
(w)(m)(m)	2.7%	0.7%	1.3%
(w)(m)(e)	3.8%	1.7%	2.8%
(w)(e)(e)	2.8%	0.9%	1.5%
(m)(m)(m)	0.6%	0.2%	0.4%
(m)(m)(e)	0.7%	0.6%	1.1%
(m)(e)(e)	0.6%	0.7%	1.2%
(e)(e)(e)	0.7%	0.2%	0.4%
four or more bonds	65.1%	4.9%	8.0%

solvent, the solvent activity is related to this osmotic pressure by

$$\ln a_{\mathrm{s}} = -\frac{\Pi v_{\mathrm{s}}}{RT}, \tag{14}$$

where v_{s} is the molar volume of the pure solvent at the temperature T, which is straightforwardly available from separate, standard molecular simulation runs. If an electrolyte solution is considered, the activity coefficient of the salt can be obtained by performing OPAS simulations at various salt molalities and applying the Gibbs-Duhem equation to the results for the solvent activity.

Fig. 5 shows molecular simulation results for the mean ionic activity coefficient of NaCl in aqueous solution at $T = 298.15$ K and $p = 1$ bar. Therein, results obtained by different groups, employing different computational approaches, are presented. Throughout, the molecular models by Joung and Cheatham [68] for Na^+ and Cl^- ions together with the SPC/E water model were used, and all simulation results are in mutual agreement.

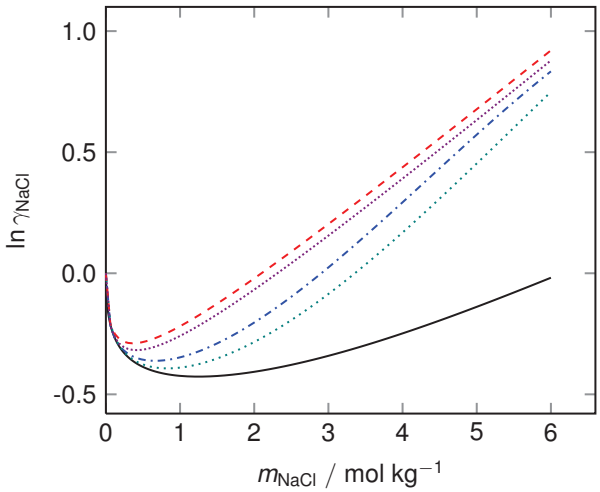

Figure 5: Mean ionic activity coefficient of NaCl over salt molality for aqueous NaCl solutions at $T = 298.15$ K and $p = 1$ bar using the SPC/E + Joung-Cheatham [68] model combination. The colored lines represent simulation results by different groups for that model combination (from top to bottom): OPAS simulations by Kohns et al. [67] (red dashed line), free energy calculations by Benavides et al. [69] (violet densely-dotted line), gradual insertion of ion pairs by Mester and Panagiotopoulos [65] (blue dashed-dotted line), and osmotic ensemble Monte Carlo simulations by Moučka et al. [66] (green dotted line). The black solid line shows the correlation to experimental data by Hamer and Wu [70].

8. Electrostatic long range corrections

The applicability of *ms*2 was extended to electrically charged molecules or ions. The long ranged intermolecular interactions are considered by two well established approaches, Ewald summation and smooth-particle mesh Ewald summation (SPME). Both are well known [37] and are thus introduced here only briefly.

In Ewald summation, the overall Coulombic potential acting in the simulation volume is determined by a sum of two terms, i.e. the short range Coulombic contribution $u^{\mathrm{c,short}}$ and the long range Coulombic contribution $u^{\mathrm{c,long}}$

$$u^{\mathrm{c}} = u^{\mathrm{c,short}} + u^{\mathrm{c,long}} \ . \tag{15}$$

The separation into these terms is achieved by the introduction of a fictive charge density function $\rho^{\mathrm{screen}}(\boldsymbol{r})$, which acts in the entire simulation volume.

Following the Ewald summation approach, for any configuration of molecules in the simulation volume, each point charge of magnitude q_l in the simulation volume is superimposed with one countercharge of magnitude $-q_l$. Due to the presence of this superimposed charge, the interaction potential decays to zero within a distance that is reasonably short for molecular simulation and can, hence, be explicitly considered by the short term contribution of the Ewald summation.

The second term in Eq. (15) determines the contribution that was subtracted due to the introduction of the fictive charge density function. This term cannot be determined explicitly by the evaluation of pairwise interactions, since the Coulombic potential at $r_{lm} = V^{1/3}/2$, i.e. the largest distance accessible in the molecular simulation volume V, is still a substantial part of the potential energy of the system. In Ewald summation, this contribution is determined in Fourier space from the negative charge distribution function $-\rho^{\mathrm{screen}}(\boldsymbol{r})$. Since $\rho^{\mathrm{screen}}(\boldsymbol{r})$ depends on the molecular positions $\boldsymbol{r}$, a Fourier transformation is performed for every configurational change in the simulation volume.

The SPME is widely considered as an improved Ewald summation method. In this approach, the concept of splitting the long range charge-charge interactions is fully employed. The difference between both methods lies only in the calculation of the long range contributions. In the SPME approach, the electrostatic field of ions in the simulation volume is described by a spline function with a given functional form. For this given spline equation, the Fourier transformation is known and has, hence, not to be determined in each simulation step. This accelerates the calculation of the long range contribution and reduces simulation effort and time. The accuracy of the SPME results is assumed to be equivalent to the Ewald approach. In *ms*2 release 3.0, the SPME method was implemented in its original form wherein the splines are evaluated for all particles at a time, making it specifically useful for MD simulations. Typical systems show an improved performance of up to 21% in total computational time.

9. Vectorization

For the vectorization of loops it is essential how accessed data are distributed in physical memory. Deducing this information automatically from the code can be very difficult for the compiler, especially if there is indirection. In *ms*2, many arrays are accessed via array pointers and the order of the underlying data is therefore obfuscated. The most relevant information is wether or not data are contiguous in physical memory. To provide this information to the compiler directly, array pointers can be given the attribute "contiguous". The Fortran standard specifies: *"The contiguous attribute specifies that [...] an array pointer can only be pointer associated with a contiguous target."* This means that the array elements of a contiguous array pointer are not

separated by any other data, potentially enabling a higher degree of vectorization. In *ms*2 release 3.0, array pointers associated with contiguous targets have been given the contiguous attribute. The sequential performance gains and the resulting parallel performance gains achieved with this optimization depend on the specific simulation scenario, but are on average very significant. For MC and MD simulations, a suite of simulations was executed to evaluate the performance gains. The suite covered the NpT and NVT ensembles, thermodynamic integration, Widom test particle insertion, different thermostats, different pure fluids and mixtures. A total of 71 simulation runs was performed. For MC simulations, the observed average reduction of runtime was more than 7% and for MD simulations more than 20%.

10. Parallel ensemble calculations

*ms*2 was already parallelized in its initial version for distributed memory architectures using the message passing interface (MPI) [9, 10]. The present release 3.0 adds an additional level of parallelization for MD simulations. Different ensembles are independent of each other in the sense that sampling different state points can be done concurrently. To achieve this, the processing elements (PE) are split in disjunct groups and each group computes the ensembles assigned to it. To enable this feature, the *mpiEnsembleGroups* option was introduced in the input file. The default (if not set or set to 0) is to use a single ensemble group. A value of 1 will enable the new feature and automatically set the number of ensemble groups to the minimum of the number of ensembles and the number of PE, i.e. *mpiEnsembleGroups*=min(ensembles, PE). Otherwise, this option will set the number of ensemble groups according to the specified integer value. The "coloring", i.e. the assignment of the PE to the ensemble groups, is done in continguous blocks. This is advantageous, e.g. compared to a round robin fashion, if the processes have to be pinned to NUMA domains.

Another change applies to the restart capability of *ms*2. A checkpoint now consists of one restart (*.rst) file for each ensemble group that can be restarted with the *ms*2 -r (or --restart) option.

To illustrate the new feature, a MD program execution for three ensembles with 24 processes will serve as an example. Without the *mpiEnsembleGroups* option (or with *mpiEnsembleGroups*=0) all 24 PE will in parallel calculate the first time step of the first ensemble, then the first time step of the second ensemble and finally the first time step of the third one, before the second time step is handled accordingly. Setting *mpiEnsembleGroups*=1 is equivalent to *mpiEnsembleGroups*=3 in this example and three groups of eight PE each will calculate one ensemble in parallel concurrently. The user may also directly specifiy the number of groups, but has to be aware that in this example with *mpiEnsembleGroups*=2 one group will have to process two ensembles, while the other one only processes a single one.

*ms*2 creates a MPI communicator for each of the ensemble groups with the MPI_Comm_Split command, cf. Figure 6. Another MPI communicator contains the root processes of all groups (Communicator_R including subcommunicators rank 0 processes) to ease collective communication on a higher level among the groups.

Even if the computation of different ensembles is embarrassingly parallel, an interaction between the different ensemble groups through communicators remains, e.g. when the program receives a signal to write a checkpoint and terminate. This signal may be received from an arbitrary single PE

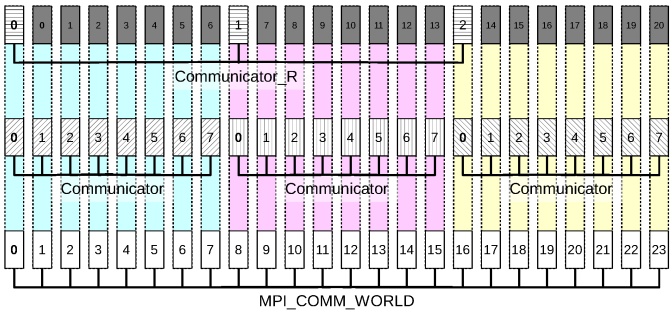

Figure 6: MPI ranks for 24 PE in 3 MPI groups, indicating communicator hierarchy of $ms2$.

(or non-isochronic from multiple PE). In the preceding releases of $ms2$, where ensembles were calculated consecutively with the single global communicator MPI_COMM_WORLD, a MPI_Allreduce spreads this information among all processes after every MD time step. This is still the case within the ensemble groups. A collective communication of all PE beyond ensemble group boundaries for every MD time step would implicitly synchronize all ensemble computations. However, this is not satisfactory because computationally less intensive ensembles would be forced to wait for slower ones to conclude their time step. The present implementation uses non-blocking communication between the ensemble group roots through Communicator_R to avoid this problem.

Before any MD time step iterations start, the root process executes a MPI_Irecv call to receive a potential terminate message, whereas all other processes execute a MPI_Ibcast to obtain a potential terminate message from the root, cf. Figure 7. During program execution, any process may trigger termination by sending a message to the root, which will then broadcast the information to notify all other processes. If no termination occurs during the iterations, the root will send a message to itself and broadcast a non-termination message to all other processes to satisfy the outstanding receive and broadcast (avoiding MPI_Cancel). To take care of several processes sending a termination message, a summation reduction determines the sum of all respective messages for the root process to receive all of them. This technique can also be used hierarchically, replacing the MPI_Allreduce call within each subcommunicator. After termination, every ensemble group writes its own restart (*.rst) file.

Acknowledgements

The authors gratefully acknowledge financial support by BMBF under the grant “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. Furthermore, we gratefully acknowledge the Paderborn Center for Parallel Computing (PC^2) for the generous allocation of computer time on the OCuLUS cluster. The present research was conducted under the auspices of the Boltzmann-Zuse Society of Computational Molecular Engineering (BZS). H.H. and M.K. acknowledge support of this work by DFG under a Reinhart Koselleck grant.

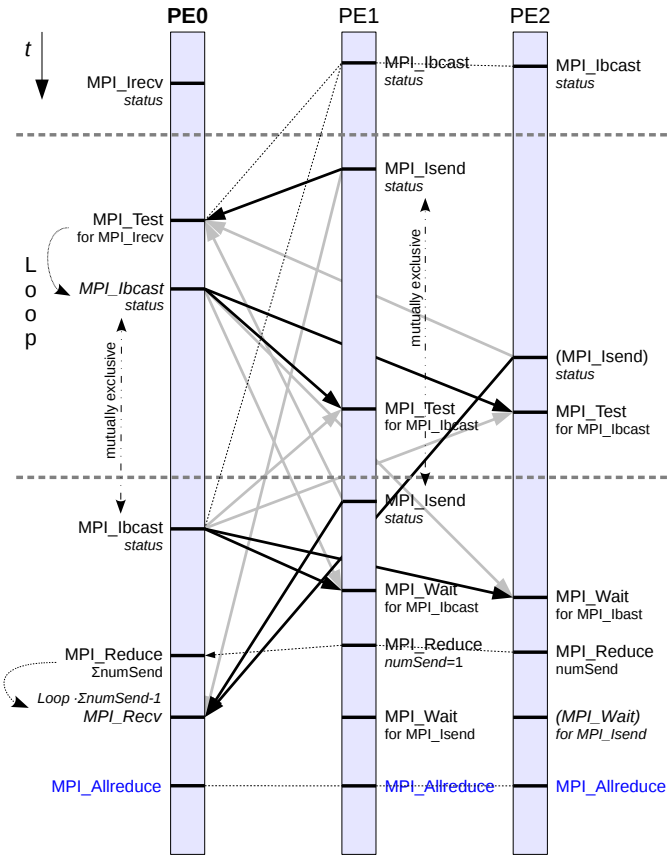

Figure 7: Communication to distribute status information within a MD time step avoiding barriers. Thick solid arrows indicate communication among PE (grey arrows for alternative possible receiving points). Dotted lines between the PE indicate collective communication implicating a barrier. For mutually exclusive commands, the second one is only executed if the first one was not triggered.

[1] P. Ungerer, C. Nieto-Draghi, B. Rousseau, G. Ahunbay, V. Lachet, J. Mol. Liq. 134 (2007) 71–89.

[2] J. C. Palmer, P. G. Debenedetti, AIChE J. 61 (2015) 370–383.

[3] A. Heinecke, W. Eckhardt, M. Horsch, H.-J. Bungartz, Supercomputing for Molecular Dynamics Simulations, Springer Verlag, Heidelberg, 2015.

[4] A. Köster, T. Jiang, G. Rutkai, C. W. Glass, J. Vrabec, Fluid Phase Equilib. 425 (2016) 84–92.

[5] C. Valeriani, R. J. Allen, M. J. Morelli, D. Frenkel, P. R. ten Wolde, J. Chem. Phys. 127 (2007) 114109.

[6] D. R. Glowacki, E. Paci, D. V. Shalashilin, J. Chem. Theory Comput. 7 (2011) 1244–1252.

[7] S. L. Meadley, F. A. Escobedo, J. Chem. Phys. 137 (2012) 074109.

[8] URL http://www.ms-2.de/.

[9] 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) 2350–2367.

[10] C. W. Glass, S. Reiser, G. Rutkai, S. Deublein, A. Köster, G. Guevara-Carrion, A. Wafai, M. Horsch, M. Bernreuther, T. Windmann, H. Hasse, J. Vrabec, Comput. Phys. Commun. 185 (2014) 3302–3306.

[11] C. Nieto-Draghi, P. Bonnaud, P. Ungerer, J. Phys. Chem. C 111 (2007) 15942–15951.

[12] E. El Ahmar, B. Creton, A. Valtz, C. Coquelet, V. Lachet, D. Richon, P. Ungerer, Fluid Phase Equilib. 304 (2011) 21–34.

[13] Y.-L. Huang, M. Heilig, H. Hasse, J. Vrabec, AIChE J. 57 (2011) 1043–1060.

[14] M. H. Ketko, G. Kamath, J. J. Potoff, J. Phys. Chem. B 115 (2011) 4949–4954.

[15] S. Reiser, M. Horsch, H. Hasse, J. Chem. Eng. Data 60 (2015) 1614–1628.

[16] S. Werth, K. Stöbener, P. Klein, K.-H. Küfer, M. Horsch, H. Hasse, Chem. Eng. Sci. 121 (2015) 110–117.

[17] R. Srivastava, H. Docherty, J. K. Singh, P. T. Cummings, J. Phys. Chem. C 115 (2011) 12448–12457.

[18] L. Lu, S. Wang, E. A. Müller, W. Cao, Y. Zhu, X. Lu, G. Jackson, Fluid Phase Equilib. 362 (2014) 227–234.

[19] A. Sharma, S. Namsani, J. K. Singh, Mol. Sim. 41 (2015) 414–422.

[20] J. L. Kern, T. J. Flynn, Z. Wang, W. H. Thompson, B. B. Laird, Fluid Phase Equilib. 411 (2016) 81–87.

[21] J. Muscatello, F. Jaeger, O. K. Matar, E. A. Müller, Appl. Mater. Interf. 8 (2016) 12330–12336.

[22] B. R. Brooks, C. L. Brooks, A. D. Mackerell, L. Nilsson, R. J. Petrella, B. Roux, Y. Won, G. Archontis, C. Bartels, S. Boresch, A. Caflisch, L. Caves, Q. Cui, A. R. Dinner, M. Feig, S. Fischer, J. Gao, M. Hodoscek, W. Im, K. Kuczera, T. Lazaridis, J. Ma, V. Ovchinnikov, E. Paci, R. W. Pastor, C. B. Post, J. Z. Pu, M. Schaefer, B. Tidor, R. M. Venable, H. L. Woodcock, X. Wu, W. Yang, D. M. York, M. Karplus, J. Comp. Chem. 30 (2009) 1545–1614.

[23] I. T. Todorov, W. Smith, K. Trachenko, M. T. Dove, J. Mater. Chem. 16 (2006) 1911–1918.

[24] H.-J. Limbach, A. Arnold, B. A. Mann, C. Holm, Comp. Phys. Comm. 174 (2006) 704–727.

[25] M. Lagache, P. Ungerer, A. Boutin, A. Fuchs, Phys. Chem. Chem. Phys. 3 (2001) 4333–4339.

[26] M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess, E. Lindahl, SoftwareX 1 (2015) 19–25.

[27] J. Roth, F. Gähler, H.-R. Trebin, Int. J. Mod. Phys. C 11 (2000) 317–322.

[28] S. Plimpton, J. Comp. Phys. 117 (1995) 1–19.

[29] C. Niethammer, S. Becker, M. Bernreuther, M. Buchholz, W. Eckhardt, A. Heinecke, S. Werth, H.-J. Bungartz, C. W. Glass, H. Hasse, J. Vrabec, M. Horsch, J. Chem. Theory Comput. 10 (2014) 4455–4464.

[30] J. C. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. D. Skeel, L. Kale, K. Schulten, J. Comp. Chem. 26 (2005) 1781–1802.

[31] P. Ren, J. W. Ponder, J. Phys. Chem. B 107 (2003) 5933–5947.

[32] M. G. Martin, Mol. Sim. 39 (2013) 1212–1222.

[33] M. Thol, G. Rutkai, A. Köster, R. Lustig, R. Span, J. Vrabec, J. Phys. Chem. Ref. Data 45 (2016) 023101.

[34] M. Hülsmann, T. Köddermann, J. Vrabec, D. Reith, Comput. Phys. Commun. 181 (2010) 499–513.

[35] M. Hülsmann, J. Vrabec, A. Maaß, D. Reith, Comput. Phys. Commun. 181 (2010) 887–905.

[36] K. Stöbener, P. Klein, S. Reiser, M. Horsch, K.-H. Küfer, H. Hasse, Fluid Phase Equilib. 373 (2014) 100–108.

[37] D. Frenkel, B. Smit, Understanding Molecular Simulation: From Algorithms to Applications, Academic Press, Elsevier, San Diego, 2002.

[38] R. Lustig, J. Chem. Phys. 109 (1998) 8816–8828.

[39] T. Kristóf, G. Rutkai, L. Merényi, J. Liszi, Mol. Phys. 103 (2005) 537–545.

[40] H. C. Andersen, J. Chem. Phys. 72 (1980) 2384–2393.

[41] J. Vrabec, J. Stoll, H. Hasse, J. Phys. Chem. B 105 (2001) 12126–12133.

[42] H. Flyvbjerg, H. G. Petersen, J. Chem. Phys. 91 (1989) 461–466.

[43] R. Span, Multiparameter Equations of State: An Accurate Source of Thermodynamic Property Data, Springer Verlag, Berlin, 2000.

[44] R. Lustig, Mol. Sim. 37 (2011) 457–465.

[45] R. Lustig, Mol. Phys. 110 (2012) 3041–3052.

[46] B. Widom, J. Chem. Phys. 39 (1963) 2808–2812.

[47] T. Kristóf, G. Rutkai, Chem. Phys. Lett. 445 (2007) 74–78.

[48] R. Krishna, J. M. van Baten, Ind. Eng. Chem. Res. 44 (2005) 6939–6947.

[49] F. H. Stillinger, Science 209 (1980) 451–457.

[50] R. H. Henchman, S. J. Irudayam, J. Phys. Chem. B 114 (2010) 16792–16810.

[51] E. Arunan, G. R. Desiraju, R. A. Klein, J. Sadlej, S. Scheiner, I. Alkorta, D. C. Clary, R. H. Crabtree, J. J. Dannenberg, P. Hobza, H. G. Kjaergaard, A. C. Legon, B. Mennucci, D. J. Nesbitt, Pure Appl. Chem. 83 (2011) 1637–1641.

[52] J. Thar, B. Kirchner, J. Phys. Chem. A 110 (2006) 4229–4237.

[53] M. Haughney, M. Ferrario, I. R. McDonald, J. Phys. Chem. 91 (1987) 4934–4940.

[54] S. Reiser, N. McCann, M. Horsch, H. Hasse, J. Supercrit. Fluids 68 (2012) 94–103.

[55] A. Luzar, D. Chandler, J. Chem. Phys. 98 (1993) 8160–8173.

[56] S. Chowdhuri, A. Chandra, J. Phys. Chem. B 110 (2006) 9674–9680.

[57] A. Luzar, D. Chandler, Nature 379 (1996) 55–57.

[58] J. R. Choudhuri, A. Chandra, J. Chem. Phys. 141 (2014) 134703.

[59] G. Guevara-Carrion, C. Nieto-Draghi, J. Vrabec, H. Hasse, J. Phys. Chem. B 112 (2008) 16664–16674.

[60] L. Saiz, J. A. Padró, E. Guàrdia, J. Phys. Chem. B 101 (1997) 78–86.

[61] W. Xu, J. Yang, Y. Hu, J. Phys. Chem. B 113 (2009) 4781–4789.

[62] D. Bhandary, S. Khan, J. K. Singh, J. Phys. Chem. C 118 (2014) 6809–6819.

[63] M. Kreitmeir, H. Bertagnolli, J. J. Mortensen, M. Parrinello, J. Chem. Phys. 118 (2003) 3639–3645.

[64] F. Moučka, I. Nezbeda, W. R. Smith, J. Chem. Phys. 139 (2013) 124505.

[65] Z. Mester, A. Z. Panagiotopoulos, J. Chem. Phys. 142 (2015) 044507.

[66] F. Moučka, I. Nezbeda, W. R. Smith, J. Chem. Theory Comput. 11 (2015) 1756–1764.

[67] M. Kohns, S. Reiser, M. Horsch, H. Hasse, J. Chem. Phys. 144 (2016) 084112.

[68] I. S. Joung, T. E. Cheatham, J. Phys. Chem. B 112 (2008) 9020–9041.

[69] A. L. Benavides, J. L. Aragones, C. Vega, J. Chem. Phys. 144 (2016) 124504.

[70] W. J. Hamer, Y. Wu, J. Phys. Chem. Ref. Data 1 (1972) 1047–1100.