

PetaFLOP Molecular Dynamics for Engineering Applications



Philipp Neumann, Nikola Tchipev, Steffen Seckler, Matthias Heinen,
Jadran Vrabec and Hans-Joachim Bungartz

Abstract Molecular dynamics (MD) simulations enable the investigation of multicomponent and multiphase processes relevant to engineering applications, such as droplet coalescence or bubble formation. These scenarios require the simulation of ensembles containing a large number of molecules. We present recent advances within the MD framework *ls1 mardyn* which is being developed with particular regard to this class of problems. We discuss several OpenMP schemes that deliver optimal performance at node-level. We have further introduced nonblocking communication and communication hiding for global collective operations. Together with revised data structures and vectorization, these improvements unleash PetaFLOP performance and enable multi-trillion atom simulations on the HLRS supercomputer Hazel Hen. We further present preliminary results achieved for droplet coalescence scenarios at a smaller scale.

P. Neumann (✉)

Department of Informatics, Universität Hamburg, Bundesstr. 45a, 20146 Hamburg, Germany
e-mail: philipp.neumann@uni-hamburg.de

N. Tchipev · S. Seckler · H.-J. Bungartz

Department of Informatics, Technical University of Munich, Boltzmannstr. 3,
85748 Garching, Germany
e-mail: tchipev@in.tum.de

S. Seckler

e-mail: seckler@in.tum.de

H.-J. Bungartz

e-mail: bungartz@in.tum.de

M. Heinen · J. Vrabec

Thermodynamics and Process Engineering, Technical University of Berlin,
Ernst-Reuter-Platz 1, 10587 Berlin, Germany
e-mail: heinen@tu-berlin.de

J. Vrabec

e-mail: vrabec@tu-berlin.de

© Springer Nature Switzerland AG 2019

W. E. Nagel et al. (eds.), *High Performance Computing in Science
and Engineering '18*, https://doi.org/10.1007/978-3-030-13325-2_25

1 Introduction

Molecular dynamics (MD) simulations have become a valuable tool for energy applications. They enable the sampling of equations of state [10], the investigation of droplet coalescence [16] and bubble formation [11] at the nanoscale, or the investigation of interfacial flows [12]. Many of the aforementioned scenarios rely on the simulation of very large molecular systems; the single molecules may, fortunately, have rather simplistic shapes and sizes. A typical example is given by a mixture of acetone (that can be modeled by four Lennard-Jones (LJ) sites, a dipole and a quadrupole) and nitrogen (two Lennard-Jones sites with a quadrupole) which is frequently used to model fuel injection-like conditions in thermodynamic laboratories. In a long-term interdisciplinary effort of computer scientists and mechanical engineers, the MD framework *ls1 mardyn* has evolved over the last decade to investigate such large systems of small molecules [13]. *ls1 mardyn* has been used in various studies [20] and has been continuously extended to optimally exploit current HPC architectures [5, 17, 19].

In the following, we detail recent developments within the framework to achieve optimal performance at node and multi-node level. After introducing the actual problem setting of short-range molecular dynamics, related work and the original implementation of *ls1 mardyn* in Sect. 2, we provide an overview of OpenMP schemes that are available in *ls1 mardyn* in Sect. 3. Improvements to MPI communication are introduced in Sect. 4. We report performance results for extreme-scale experiments in Sect. 5 and preliminary results on droplet coalescence in Sect. 6. We close with a summary and an outlook to future work in Sect. 7. Scalability and performance results have recently been published in [18].

2 Short-Range Molecular Dynamics

2.1 Theory

In short-range MD, Newton's equations of motion are numerically solved [15]. In the following, considerations are restricted to small molecules. Due to their negligible conformational changes, molecules mainly undergo translational or rotational motion; both are included in the equations of motion and are solved simultaneously in *ls1 mardyn* using a leapfrog time integrator, without the need for iterative procedures (such as the SHAKE algorithm) to handle geometric constraints [15].

Molecules interact via force fields. In short-range MD, arising forces are only explicitly accounted for if the distance between two considered molecules is below a specified cut-off radius r_c . There are basically two variants to efficiently implement the cut-off condition: linked cells and Verlet lists [15]. Both methods turn the actual molecule-molecule interaction complexity from $O(N^2)$ to $O(N)$. *ls1 mardyn* makes use of the linked cell approach: a Cartesian grid with cell sizes $\geq r_c$ is introduced and

covers the computational domain. The molecules are sorted into the cells. Molecular interactions only need to be considered for molecules that reside within the same cell or in neighboring cells.

Various interaction potentials have been proposed to model LJ molecular behavior. The simulations reported in this contribution make use of the LJ potential [15]

$$U(r_{ij}) = 4\varepsilon \left(\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^6 \right), \quad (1)$$

with molecule-dependent parameters σ and ε and the distance of molecules i and j given by r_{ij} , as well as the electrostatic quadrupole potential for the interaction of two point-quadrupoles [7]. The force calculation is typically by far the most expensive step in MD and often contributes $\geq 90\%$ to the overall compute time.

2.2 Related Work

HPC and Related MD Implementations

Various packages efficiently and flexibly implement (short-range) molecular dynamics algorithms, with the most popular ones given by Gromacs,¹ LAMMPS² and NAMD.³ Gromacs leverages particularly GPUs but also supports OpenMP and large-scale MPI parallelism, and it also exploits SIMD instructions via a new particle cluster-based Verlet list method [1, 14]. A LAMMPS-based short-range MD implementation for host-accelerator systems is reported in [2] with speedups for LJ scenarios of 3–4. A pre-search process to improve neighbor list performance at SIMD level and an OpenMP slicing scheme are presented in [9, 21]. The arising domain slices, however, need to be thick enough, to actually boost performance at shared-memory level. This restricts the applicability of the method to rather large (sub-)domains per process.

ls1 mardyn

An approach to efficient vectorization built on top of the linked cell data structure within *ls1 mardyn* is presented for single- [4] and multi-site⁴ molecules [3]. This method, combined with a memory-efficient storage, compression and data management scheme [6], allowed for a four-trillion atom simulation in 2013 on the supercomputer SuperMUC, phase 1 [5]. A multi-dimensional, OpenMP-based coloring approach that operates on the linked cells is provided in [19]. The method has been evaluated on both Intel Xeon and Intel Xeon Phi architectures and exhibits good scalability up to the hyperthreading regime. *ls1 mardyn* further supports load balancing. It

¹www.gromacs.org.

²www.lammps.org.

³<http://www.ks.uiuc.edu/Research/namd/>.

⁴Molecules that consist of several interaction sites, e.g. two LJ sites.

uses k-d trees for this purpose. Recently, this approach has been employed to balance computational load on heterogeneous architectures [17]. A detailed overview of the original release of *ls1 mardyn* is provided in [13]. Various applications from process and energy engineering, including several case studies that exploit *ls1 mardyn*, are discussed in [20].

3 OpenMP Schemes

ls1 mardyn has recently been extended by two novel OpenMP parallelization schemes for the force calculation, complementing a preceding implementation. All three schemes are depicted in Fig. 1. The schemes **c08** and **c04**, depicted in Fig. 1a, c, employ coloring to the linked cells and thus enable parallelism at shared-memory level, avoiding race conditions when accumulating all force contributions per molecule. The scheme **c08** comes with finer granularity at cell level, but at the cost of more read/write operations per molecule. The scheme **s1i** basically arranges all cells in a linear order and cuts this linearized cell accumulation into as many slices as threads are available. The scheme works particularly well for large domains since synchronization between the threads, achieved by a locking mechanism, is minimal in this case. Details on the schemes can be found in [18, 19].

We evaluated the schemes in single-site Lennard-Jones experiments on Hazel Hen, cf. Fig. 2, with 47×10^6 molecules, cut-off radius $r_c/\sigma = 3.5$ and density $\rho\sigma^3 = 0.78$. Performance is given in million molecule updates per second (MMUPS) per thread in this figure. All schemes exhibit good scalability at node level, showing some advantages of **s1i** over **c04** over **c08**. This, however, is strongly scenario-specific. For example, we encountered better performance of **c04** in a two-phase vapor-liquid scenario (that is, a scenario with high load imbalances because one region is densely packed with molecules (liquid phase) and another region where molecules are sparsely present (vapor phase)).

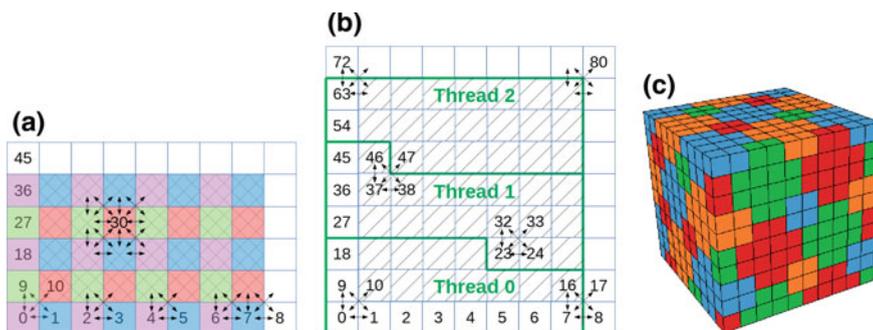
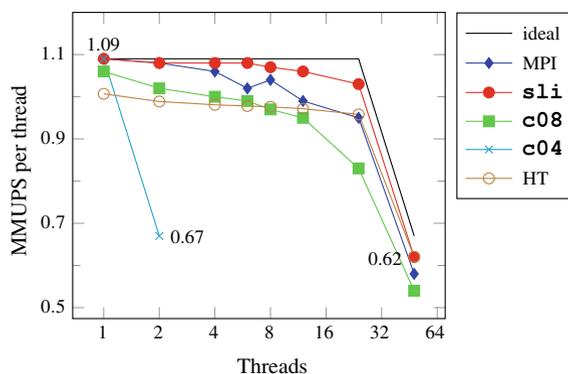


Fig. 1 OpenMP schemes integrated in *ls1 mardyn*: **a** **c08** coloring, **b** **s1i**, **c** **c04** coloring

Fig. 2 Strong scaling on one node of Hazel Hen. HT denotes two threads pinned on the same core running `s1i`. Turbomode was switched off



4 Communication Hiding

MD processes such as droplet coalescence operate on long time scales, yielding the requirement of (hundreds of) millions of time steps that have to be executed. Due to the sequential processing of the time steps and with the time scale (nonlinearly) increasing with the number of molecules, the size of the overall MD system becomes limited and improving “time-to-solution” is of utmost importance. As a consequence, codes such as *ls1 mardyn* have to feature optimal strong scalability. Therefore, we have improved MPI communication in *ls1 mardyn*.

Communication Between Neighboring Subdomains

ls1 mardyn applies a domain decomposition—regular Cartesian domain decomposition or decomposition based on k-d trees [17]—to split the MD domain and distribute computational load. Molecules are sent from one subdomain to another (a) if they have propagated to the other subdomain within a time step or (b) if they are located close to the other subdomain and are thus required for local force calculations. We have reduced the size of the messages that are exchanged between subdomains taking into account (a) and (b): if a molecule is required due to (b), we only send the molecular data that is required for the force calculation (i.e., the molecule position data). In case (a), we send the entire molecule data set to the other subdomain. All required molecular data of type (a) and (b) are collected in one buffer and sent in one message to a neighboring subdomain. Different communication schemes are supported and have been investigated in *ls1 mardyn*, including fully asynchronous data transfer, which is supported for all spatial decomposition schemes, as well as schemes that make a trade-off between the number of communication steps and the number of communication neighbors. The load balancing has been enhanced to support almost-arbitrary heterogeneous loads by using in-situ time measurements.

Nonblocking Collectives

Collectives are required to gather statistical properties from the MD system, such as temperature or energy values. These properties vary rather slowly over time. We

exploit this “feature” and have incorporated MPI-3 based nonblocking collectives into *ls1 mardyn*, which assemble statistical properties by relying on data from a preceding time step. This allows to hide global reduction operations behind actual MD time step evaluations.

5 Scalability and Performance at Large Scale

We investigated both weak and strong scalability on Hazel Hen in various scenarios, cf. Fig. 3. The molecular ensemble consisted of single-site LJ molecules at a liquid density $\rho\sigma^3 = 0.78$, specifying a cut-off radius $r_c/\sigma = 3.5$.

In the weak scaling scenario, we were able to simulate up to 2.1×10^{13} particles on 7168 nodes using 1 MPI rank per node and 48 OpenMP threads at node-level; the latter setting yields minimum memory requirements compared to other hybrid MPI/OMP configurations. On 7168 nodes, the scenario would correspond to a cube with a width of $11.8 \mu\text{m}$, referring to physical units for liquid xenon ($\sigma = 3.9450 \text{ \AA}$).

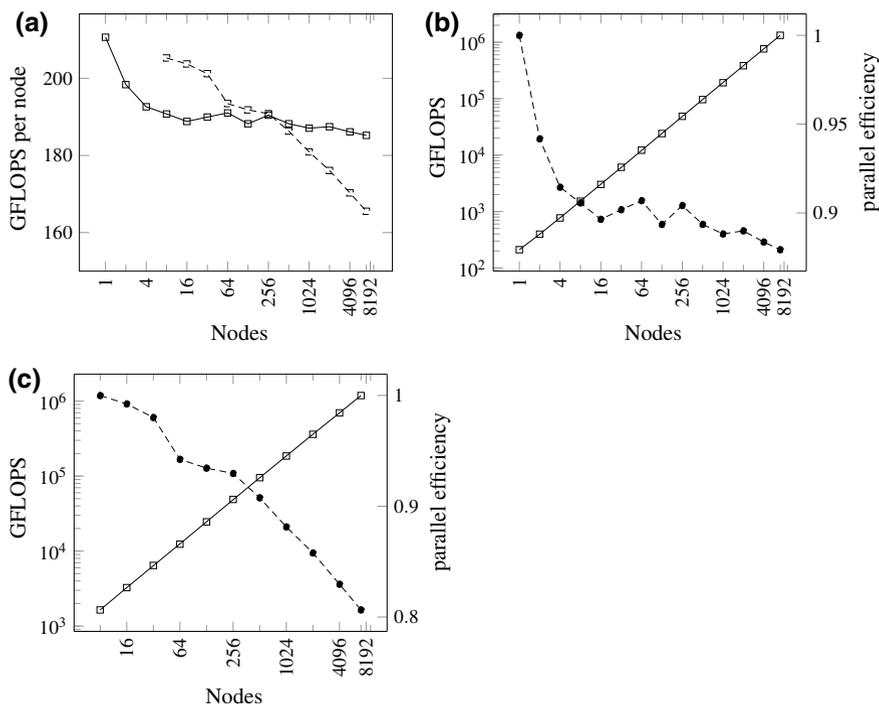


Fig. 3 Scalability experiments on up to 7168 Hazel Hen nodes. **a** GFLOPS per node in weak (solid line) and strong scaling (dashed line). **b** Total GFLOPS (solid line) and parallel efficiency (dashed line) in weak scaling. **c** Total GFLOPS and parallel efficiency in strong scaling

The weak scaling efficiency was measured to be $\geq 88\%$ (cf. Fig. 3b). A performance of 1.33 PFLOPS on 7168 nodes was achieved (9% of single precision peak of Hazel Hen), corresponding to 189×10^9 molecule updates per second.

Strong scalability was investigated using a system of 23.85 billion molecules. We achieved a performance of 1.18 PFLOPS and a parallel efficiency of 81% on 7168 nodes. In this case, we used 6 MPI ranks and 8 OpenMP threads per MPI rank which had been shown to provide slightly superior performance over the 1MPI/48OMP setting in previous tests. The simulation rate arose as 178×10^9 molecule updates per second.

6 Droplet Coalescence

Enabled by a new sampling module for recording the temporal development of a two-dimensional density field $\rho(z, r)$, preliminary results of two coalescing argon droplets were obtained from molecular dynamics simulations with *ls1 mardyn*, cf. Figs. 4 and 5. The first application area of this sampling module were rotationally

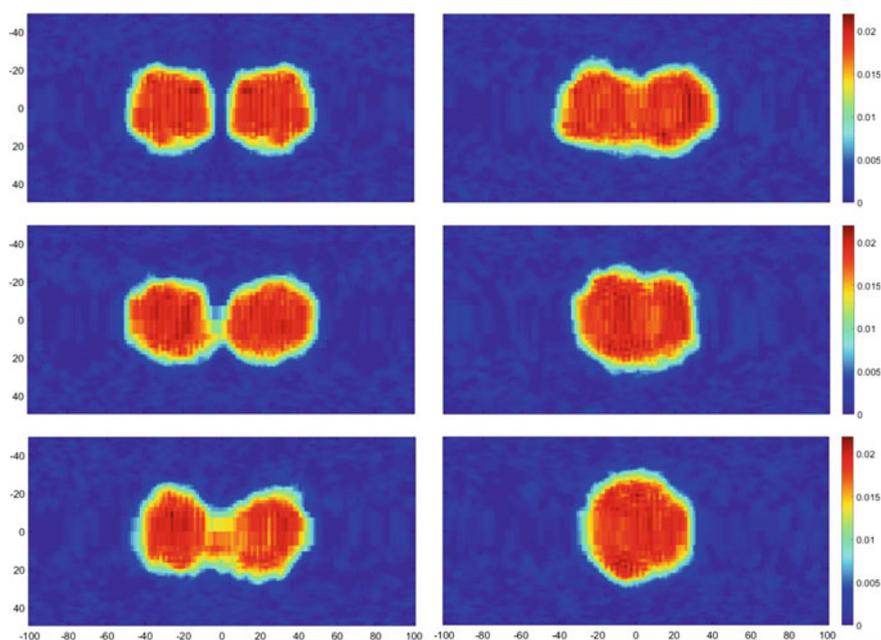


Fig. 4 Preliminary work on coalescence, specifically two argon droplets with a diameter of $d = 5$ nm and a number of $N \approx 10^3$ particles in equilibrium with their vapor at a temperature of $T = 110$ K. From top left to bottom right, the sequence shows a time period of $\Delta t \approx 0.1$ ns in total. The snapshots exhibit the two-dimensional density field $\rho(z, r)$, sampled by a new module of *ls1 mardyn* as depicted by a colormap

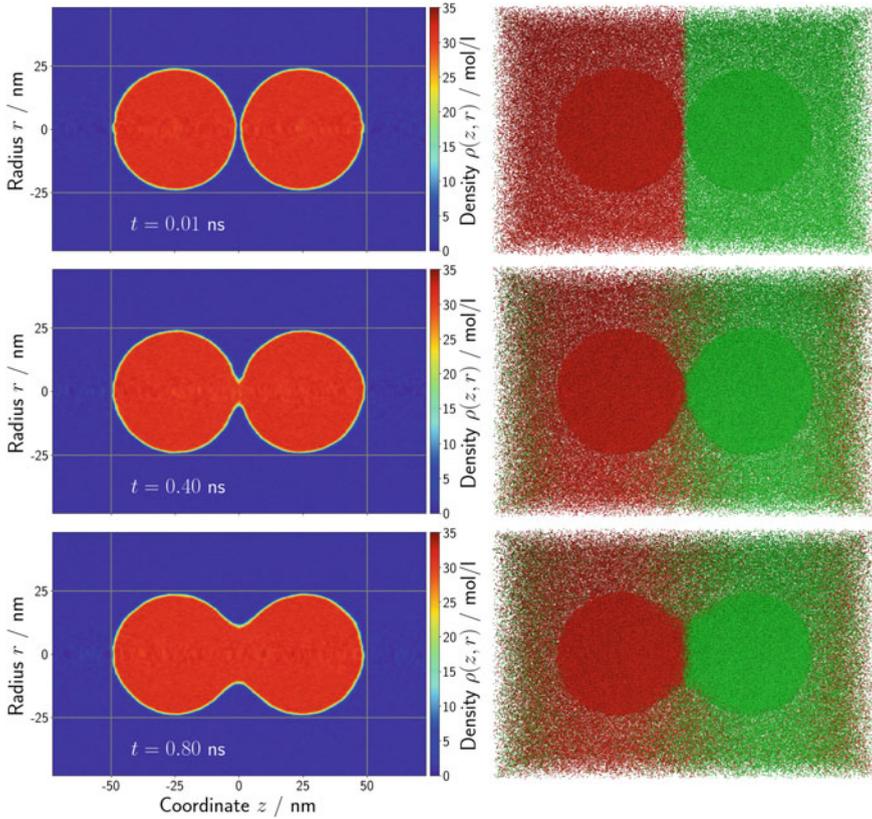


Fig. 5 Preliminary work on the subject of coalescence, specifically two argon droplets with a diameter of $d = 50$ nm containing a number of $N \approx 10^6$ particles in equilibrium with their vapor at a temperature of $T = 110$ K. The left column shows two-dimensional density fields, sampled by a new module of *lsl mardyn* as depicted by a colormap. The right column shows snapshots of the visualization by the cross-platform visualization framework *MegaMol* [8], considering every single particle of the system ($\approx 10^6$ per droplet). The colors red and green were selected to be able to observe the diffusive mixing of particles that initially were located either in or around the left or right droplet. The sequence shows a time period of $\Delta t = 0.4$ ns

symmetrical systems around the z axis. In the first test scenario, two droplets with a diameter of $d = 5$ nm containing a number of particles $N \approx 10^3$ were centered along the z axis with a mutual distance of $\delta = 1$ nm, within and in thermodynamic equilibrium with a vapor phase. The centers of mass of the droplets initially had a relative velocity of zero. The workflow to setup such a scenario is to first separately equilibrate the two phases (liquid and vapor) in a cubic volume and subsequently cut the droplets out of the liquid phase and place them into the vapor phase, to invoke a physical configuration. The evolution of the coalescence process is depicted in Fig. 4, showing a time period of $\Delta t \approx 0.1$ ns in total. Due to the short distance between the droplets of only 1 nm, the particles located at their surfaces are mutually attracted

by the opposite interface. Consequently, a liquid bridge forms within a fraction of a nanosecond. This bridge rapidly grows, driven by the surface tension. At the time instance when the thickness of the bridge is almost the same as the droplet diameter, the outer boundaries of the droplets on the left and right sides start to propagate towards the center of the connected droplets, forming a single larger one. The pictures in Fig. 4 clearly show that because of the small size of the droplets, their shape, represented by the interface, is strongly affected by fluctuations. In the field of nanotechnology, scientists might be interested in the dynamics of such small droplets. In this case molecular dynamics obviously would be the appropriate method to study these effects. The main objective of this study, however, is rather advanced. A major advantage of molecular dynamics is its sound physical background. Assumptions or approximations are limited to the molecular model, covering the force fields and hence the interactions between molecules. The system under investigation does not need any further assumptions to set up boundary conditions, e.g. at a vapor-liquid interface, like it is usually done in computational fluid dynamics (CFD). Hence, molecular dynamics can favorably serve as a benchmark for methods dealing with systems on more aggregated scales, like CFD, to validate their assumptions or chosen boundary conditions.

For studying the dynamics of coalescence processes of macroscopic droplets, and in this course to compare the results of atomistic simulations to those of continuum approaches, significantly larger droplets have to be considered. They have to be at least large enough to not be dominated by the strong fluctuations on the nano-scale. Therefore, a second scenario with by one order of magnitude larger droplets was created, i.e. with a droplet diameter of $d = 50$ nm. Since the volume and therefore also the number of particles of the droplet is proportional to the droplet diameter to the power of three, $N \propto d^3$, the larger droplets consisted of $N \approx 10^6$ particles. The results of this study are shown in Fig. 5. Obviously, the evolution of the interface's shape during the coalescence process can be visually followed very clearly, and therefore allows for quantitative comparison with other methods. These investigations are planned to be continued by varying different parameters of the molecular model, but also by variation of the thermodynamic state point, e.g. the system temperature T . Additionally, the calculations will be repeated with a further increase of the droplet size, to be able to extrapolate the results to macroscopic droplets.

7 Summary and Outlook

We have provided results from large-scale MD simulations leveraging the performance of basically the entire Hazel Hen supercomputer, achieving PetaFLOP performance for systems with billions up to 20 trillions of molecules. In the largest of our runs, we however encountered some issues in terms of performance loss by up to 50% and failing nodes. The cause of these performance decreases is under current investigation. Considering three different OpenMP schemes, we observed that one or the other scheme may be favorable in a simulation scenario. This confirms the

need for auto-tuning at OpenMP level which is part of the work plan of our current GCS-MDDC compute project at HLRS. Preliminary work on the subject of droplet coalescence showed that to meet the target of being able to compare the results of the atomistic simulations to those of methods operating on macroscopic scales, e.g. CFD, utilizing continuum approaches, the droplets have to be at least large enough so that the dynamics of the coalescence process is not dominated by the strong fluctuations on the nano-scale. Future work will focus on integrating auto-tuning into the simulation and address even larger droplet coalescence scenarios.

Acknowledgements The presented work was carried out in the scope of the Large-Scale Project *Extreme-Scale Molecular Dynamics Simulation of Droplet Coalescence*, acronym GCS-MDDC, of the Gauss Centre for Supercomputing; the authors thank Bernd Krischok, Dr.-Ing. Martin Bernreuther and Prof. Dr. Michael Resch for their support throughout the project. Financial support by the Federal Ministry of Education and Research, project *Task-based load balancing and auto-tuning in particle simulations (TaLPas)*, grant numbers 01IH16008A/B/E, is acknowledged.

References

1. M. Abraham, T. Murtola, R. Schulz, S. Páll, J. Smith, B. Hess, E. Lindahl, GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX* **1–2**, 19–25 (2015)
2. W. Brown, P. Wang, S. Plimpton, A. Tharrington, Implementing molecular dynamics on hybrid high performance computers—short range forces. *Comput. Phys. Commun.* **182**(4), 898–911 (2011)
3. W. Eckhardt, Efficient HPC implementations for large-scale molecular simulation in process engineering. Dissertation, Dr. Hut, Munich, 2014
4. W. Eckhardt, A. Heinecke, An efficient vectorization of linked-cell particle simulations, in *ACM International Conference on Computing Frontiers*, ACM, New York, NY, USA, pp. 241–243 (2012)
5. W. Eckhardt, A. Heinecke, R. Bader, M. Brehm, N. Hammer, H. Huber, H.G. Kleinhenz, J. Vrabec, H. Hasse, M. Horsch, M. Bernreuther, C. Glass, C. Niethammer, A. Bode, H.J. Bungartz, 591 TFLOPS multi-trillion particles simulation on superMUC (Springer, Berlin, Heidelberg, 2013), pp. 1–12
6. W. Eckhardt, T. Neckel, Memory-efficient implementation of a rigid-body molecular dynamics simulation, in *Proceedings of the 11th International Symposium on Parallel and Distributed Computing—ISPD 2012* (Munich, 2012). IEEE, pp. 103–110
7. C. Gray, K. Gubbins, Theory of Molecular Fluids: I: Fundamentals, in *International Series of Monogr.*, vol.1 (Oxford University Press, Oxford, 1984)
8. S. Grottel, M. Krone, C. Müller, G. Reina, T. Ertl, MegaMol—a prototyping framework for particle-based visualization. *IEEE Trans. Vis. Comput. Graph.* **21**(2), 201–214 (2015)
9. C. Hu, X. Wang, J. Li, X. He, S. Li, Y. Feng, S. Yang, H. Bai, Kernel optimization for short-range molecular dynamics. *Comput. Phys. Commun.* **211**, 31–40 (2017)
10. A. Köster, T. Jiang, G. Rutkai, C. Glass, J. Vrabec, Automated determination of fundamental equations of state based on molecular simulations in the cloud. *Fluid Phase Equilibria* **425**, 84–92 (2016)
11. K. Langenbach, M. Heilig, M. Horsch, H. Hasse, Study of homogeneous bubble nucleation in liquid carbon dioxide by a hybrid approach combining molecular dynamics simulation and density gradient theory. *J. Chem. Phys.* **148**(124702) (2018)
12. G. Nagayama, P. Cheng, Effects of interface wettability on microscale flow by molecular dynamics simulation. *Int. J. Heat Mass Transf.* **47**, 501–513 (2004)

13. C. Niethammer, S. Becker, M. Bernreuther, M. Buchholz, W. Eckhardt, A. Heinecke, S. Werth, H.J. Bungartz, C. Glass, H. Hasse, J. Vrabec, M. Horsch, ls1 mardyn: the massively parallel molecular dynamics code for large systems. *J. Chem. Theory Comput.* **10**(10), 4455–4464 (2014)
14. S. Páll, B. Hess, A flexible algorithm for calculating pair interactions on SIMD architectures. *Comput. Phys. Commun.* **184**(12), 2641–2650 (2013)
15. D. Rapaport, *The Art of Molecular Dynamics Simulation* (Cambridge University Press, Cambridge, 2004)
16. L. Rekvig, D. Frenkel, Molecular simulations of droplet coalescence in oil/water/surfactant systems. *J. Chem. Phys.* **127**(134701) (2007)
17. S. Seckler, N. Tchipev, H.J. Bungartz, P. Neumann, Load balancing for molecular dynamics simulations on heterogeneous architectures, in *2016 IEEE 23rd International Conference on High Performance Computing (HiPC)*, pp. 101–110 (2016)
18. N. Tchipev, S. Seckler, M. Heinen, J. Vrabec, F. Gratl, M. Horsch, M. Bernreuther, C. Glass, C. Niethammer, N. Hammer, B. Krischok, M. Resch, D. Kranzlmüller, H. Hasse, H.J. Bungartz, P. Neumann, TeTRiS: twenty trillion-atom simulation. *Int. J. High Perform. Comput. Appl.* (2019), <https://doi.org/10.1177%2F1094342018819741>
19. N. Tchipev, A. Wafai, C. Glass, W. Eckhardt, A. Heinecke, H.J. Bungartz, P. Neumann, Optimized Force Calculation in Molecular Dynamics Simulations for the Intel Xeon Phi (Springer International Publishing, Cham, 2015), pp. 774–785)
20. J. Vrabec, M. Bernreuther, H.J. Bungartz, W.L. Chen, W. Cordes, R. Fingerhut, C. Glass, J. Gmehling, R. Hamburger, M. Heilig, M. Heinen, M. Horsch, C.M. Hsieh, M. Hülsmann, P. Jäger, P. Klein, S. Knauer, T. Köddermann, A. Köster, K. Langenbach, S.T. Lin, P. Neumann, J. Rarey, D. Reith, G. Rutkai, M. Schappals, M. Schenk, A. Schedemann, M. Schönherr, S. Seckler, S. Stephan, K. Stöbener, N. Tchipev, A. Wafai, S. Werth, H. Hasse, Skasim—Scalable hpc software for molecular simulation in the chemical industry. *Chem. Ing. Tech.* **90**(3), 295–306 (2018)
21. X. Wang, J. Li, J. Wang, X. He, N. Nie, Kernel Optimization on Short-Range Potentials Computations in Molecular Dynamics Simulations (Springer, Singapore, 2016), pp. 269–281