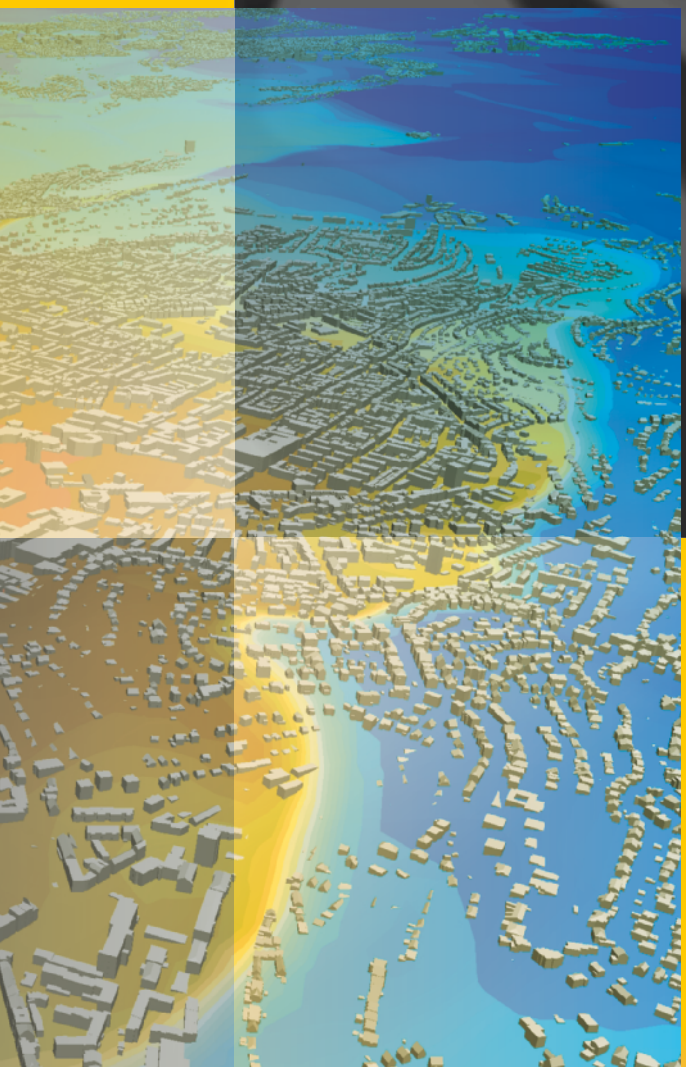
Wolfgang E. Nagel
Dietmar H. Kröner
Michael M. Resch *Editors*

High Performance Computing in Science and Engineering '20



H L R S

Springer

High Performance Computing in Science and Engineering ’20

Wolfgang E. Nagel · Dietmar H. Kröner ·
Michael M. Resch
Editors

High Performance Computing in Science and Engineering ’20

Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2020

Springer

Editors
Wolfgang E. Nagel
Zentrum für Informationsdienste und Hochleistungsrechnen (ZIH)
Technische Universität Dresden
Dresden, Germany

Dietmar H. Kröner
Abteilung für Angewandte Mathematik
Universität Freiburg
Freiburg, Germany

Michael M. Resch
Höchstleistungsrechenzentrum Stuttgart (HLRS)
Universität Stuttgart
Stuttgart, Germany

Front cover figure: The image shows the simulated NO_2 concentrations (µg/m³) during the morning peak traffic time on January 21, 2019. The viewing direction is from northeast to southwest with the Stuttgart main station in the foreground. Brownish colors indicate high concentrations of more than 120 µg/m³ while dark blue colors indicate concentrations of less than 20 µg/m³. Details can be found in "WRF Simulations to Investigate Processes Across Scales (WRFSCALE)" by Hans-Stefan Bauer, Thomas Schwitalla, Oliver Branch and Rohith Thundathil, Institute of Physics and Meteorology, University of Hohenheim, Germany, on pages 469ff.

ISBN 978-3-030-80601-9 ISBN 978-3-030-80602-6 (eBook)
https://doi.org/10.1007/978-3-030-80602-6

Mathematics Subject Classification: 65Cxx, 65C99, 68U20, 90-08, 97M50

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Contents

Computational Fluid Dynamics

Physics

Peter Nielaba

In this section, four physics projects are presented, which achieved important scientific results in 2019/20 by using Hazel Hen/Hawk at the HLRS and ForHLR II of the Steinbuch Centre.

Fascinating new results are being presented in the following pages on quantum systems (anomalous magnetic moment of the muon, photoionization of neon, ultracold Bosons' hierarchical superfluidity in a cavity, dynamics of fragmentation, entropy, angular momentum, correlations, and fluctuations of interacting bosons and fermions in one- and two-dimensional double-well potentials), on soft matter/biochemical systems (human proteasome inhibited by a covalent ligand), and on astrophysical systems (general relativistic binary merger evolutions).

Studies of the quantum systems have focused on the anomalous magnetic moment of the muon, and on the photoionization of neon, ultracold Bosons' hierarchical superfluidity in a cavity, as well as on the dynamics of fragmentation, entropy, angular momentum, correlations, and fluctuations of interacting bosons and fermions in one- and two-dimensional double-well potentials.

M. Cè, A. Gérardin, G. von Hippel, B. Hörz, R. J. Hudspith, H.B. Meyer, K. Miura, D. Mohler, K. Ottnad, S. Paul, A. Risch, T. San José, S. Schaefer, J. Wilhelm, and H. Wittig from Mainz (M.C., A.G., G.H., B.H., R.J.H., H.B.M., K. M., D.M., K.O., S.P., A.R., T.S., J.W., H.W.), Darmstadt (M.C., K.M., D.M., T.S.), CERN (M.C.), Zeuthen (A.G., S.S.), Marseille (A.G.), and Berkeley (B.H.) present interesting results obtained by their lattice QCD Monte Carlo simulations on Hazel Hen in their project *GCS-HQCD* on hadronic contributions to the anomalous magnetic moment of the muon. In particular, the authors focussed on the hadronic vacuum polarization and light-by-light scattering contributions to the muon anomalous magnetic moment, as well as on the hadronic contributions to the energy dependence of the electromagnetic coupling and the electroweak mixing angle.

P. Nielaba
Fachbereich Physik, Universität Konstanz, 78457 Konstanz, Germany
e-mail: peter.nielaba@uni-konstanz.de

Despite the important progress made in the project, the authors conclude that the results indicate that additional statistics would be important to decide on the importance of "new physics" contributions.

A. U. J. Lode, O.E. Alon, M.A. Bastarrachea-Magnani, A. Bhowmik, A. Buchleitner, L. S. Cederbaum, R. Chitra, E. Fasshauer, L. de Forges de Parny, S. K. Haldar, C. Leveque, R. Lin, L. B. Madsen, P. Molignini, L. Papariello, F. Schäfer, A. I. Strelstov, M. C. Tsatsos, and S.E. Weiner from Freiburg (A.U.J.L., M.A.B., A.B., L.F., F.S.,), Haifa (O.E.A., A.B., S.K.H.,), Aarhus (M.A.B., E.F., L.B.M.,), Heidelberg (L.S.C., A.I.S.), Zürich (R.C., R.L., P.M., L.P.,), Sophia Antipolis (L.F.,), Sonipat (S.K.H.), Wien (C.L., L.P.), Oxford (P.M.,), Basel (F.S.), Sao Paulo (M.C.T.), and Berkeley (S.E.W.) present interesting results obtained in their project *MCTDH-X* with their multiconfigurational time-dependent Hartree method for indistinguishable particles (MCTDH-X) on Hazel Hen and Hawk. In the past the authors have implemented their method to solve the many-particle Schrödinger equation for time-dependent and time-independent systems in various software packages (MCTDHB, MCTDHB-LAB, and MCTDH-X). The authors present their method and interesting new results of their investigations on the photoionization of neon, the superfluidity in Bose-Einstein condensates interacting with light, the dynamical behavior of bosons and fermions in a double well, the spectral structure and many-body dynamics of ultracold Bosons in a double-well, the universality of fragmentation and fragmented resonant tunneling in an asymmetric bosonic Josephson junction, and on the impact of the transverse direction on the many-body tunneling dynamics in a two-dimensional bosonic Josephson junction.

Studies of the soft matter/biochemical systems have focused on the human proteasome inhibited by a covalent ligand.

M. H. Kolář, L. V. Bock, and H. Grubmüller from Göttingen (M.H.K., L.V.B., H.G.) and Prague (M.H.K.) present interesting results obtained in their project *GCS-Prot* on the human proteasome inhibited by a covalent ligand by atomistic simulations with the GROMACS 2016 package and a self-compiled version on Hazel Hen. The authors carried out atomistic molecular dynamics simulations of the native and inhibited proteasomes to understand the molecular details of the inhibition, describe here the technical details of the simulations and assess the quality of the trajectories obtained. The biochemical aspects of the proteasome are under further investigation.

The studies of the astrophysical systems have focused on general relativistic binary merger evolutions.

R. Gold and L. Rezolla from Frankfurt am Main present interesting results obtained in their project *BBHDISKS* on general relativistic astrophysics and compact objects on Hazel Hen. The authors present matter evolutions in spacetime metrics of two black holes in orbit around each other. After software development and merging (BHAC/MPI-AMRVAC/Kadath), the authors are now able to evaluate

the spacetime metric to spectral accuracy on any (e.g. non-uniform) numerical grid. The authors initialized and evolved matter configurations within a binary spacetime and can now handle the nature of black hole horizons.

Hadronic Contributions to the Anomalous Magnetic Moment of the Muon from Lattice QCD

M. Cè, A. Gérardin, G. von Hippel, B. Hörz, R. J. Hudspith, H. B. Meyer, K. Miura, D. Mohler, K. Ottnad, S. Paul, A. Risch, T. San José, S. Schaefer, J. Wilhelm, and H. Wittig

Abstract The Standard Model of Particle Physics constitutes a highly successful theoretical framework for the treatment of the strong, electromagnetic and weak interactions. Still, it fails to provide explanations for dark matter or the abundance of matter over antimatter in the universe. A promising hint for physics beyond the Standard Model is provided by the persistent tension of 3.7 standard deviations between the theoretical estimate for the muon anomalous magnetic moment, $a_\mu \equiv \frac{1}{2}(g-2)_\mu$, and its direct measurement. With the advent of new and more precise measurements, the precision of the theoretical estimate, which is dominated by effects of the strong interaction, must be increased. In our project we compute a variety of hadronic contributions to precision observables, using the first-principles method of Lattice QCD. In particular, we focus on the hadronic vacuum polarisation and light-by-light scattering contributions to a_μ, as well as the hadronic contributions to the energy dependence of the electromagnetic coupling and the electroweak mixing angle.

M. Cè
Theoretical Physics Department, CERN, Geneva, Switzerland

A. Gérardin
Aix Marseille Univ, Université de Toulon, CNRS, CPT, Marseille, France

G. von Hippel · R. J. Hudspith · H. B. Meyer · K. Ottnad · S. Paul · J. Wilhelm · H. Wittig (✉)
Institut für Kernphysik and PRISMA+ Cluster of Excellence, Universität Mainz,
Becher Weg 45, 55099 Mainz, Germany
e-mail: hartmut.wittig@uni-mainz.de

B. Hörz
Nuclear Science Division, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

K. Miura · D. Mohler · T. San José
Helmholtz Institut Mainz, Johannes Gutenberg Universität and GSI Helmholtzzentrum für Schwerionenforschung, Darmstadt, Germany

A. Risch · S. Schaefer
John von Neumann-Institut für Computing, Deutsches Elektronen-Synchrotron, DESY,
Platanenallee 6, 15768 Zeuthen, Germany

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_1

1 Introduction

The Standard Model of Particle Physics (SM) provides a quantitative and precise description of the properties of the known constituents of matter in terms of a uniform theoretical formalism. However, despite its enormous success, the SM fails at explaining some of the most pressing problems in particle physics, such as the nature of dark matter or the asymmetry between matter and antimatter. The world-wide quest for discovering physics beyond the SM involves several different strategies, namely (1) the search for new particles and interactions that are not described by the SM, (2) the search for the enhancement of rare processes by new interactions, and (3) the comparison of precision measurements with theoretical, SM-based predictions of the same quantity. These complementary activities form an integral part of the future European strategy for particle physics [1].

Precision observables, such as the anomalous magnetic moment of the muon, a_μ, have attracted a lot of attention recently, chiefly because of the persistent tension of 3.7 standard deviations between the direct measurement of a_μ and its theoretical prediction. As the community prepares for the announcement of a new direct measurement by the E989 experiment at Fermilab, which is supposed to increase the precision by up to a factor of four in the long run, the precision of the theoretical prediction has to be increased as well. Since the main uncertainties in the SM prediction are associated with the effects from the strong interaction, current efforts are focussed on quantifying the contributions from hadronic vacuum polarisation (HVP) and hadronic light-by-light scattering (HLbL). This has also been emphasised in a recent white paper [2] in which the status of the theoretical prediction is reviewed.

Our project is focussed on calculations of the hadronic contributions to the muon anomalous magnetic moment from first principles, using the method of Lattice QCD. To this end, we perform calculations of the HVP contribution at the physical value of the pion mass, in order to reduce systematic errors. Furthermore, we perform calculations of the transition form factor for the process $\pi^0 \to \gamma\gamma$, which plays an important role for quantifying the HLbL contribution. We have also developed a new formalism for the direct calculation of the HLbL contribution, which we put to a first test as part of this project.

The HVP contribution to the muon anomalous magnetic moment is closely linked to the hadronic effects that modify the value of the electromagnetic coupling, $\Delta\alpha$. Since $\Delta\alpha$ depends on other SM parameters such as the mass of the W-boson, a precise determination provides important information for precision tests of the SM. Finally, we also compute the hadronic contributions to the "running" of the electroweak mixing angle, a precision observable which is particularly sensitive to the effects of physics beyond the SM in the regime of low energies.

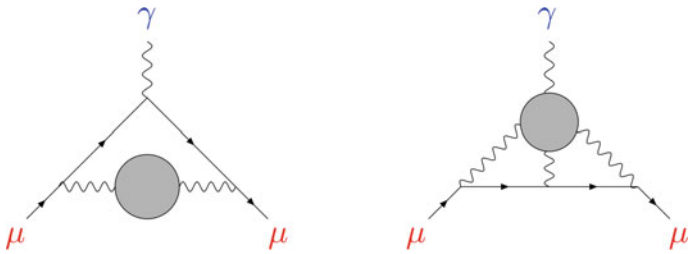


Fig. 1 Diagrams for the hadronic vacuum polarisation (left) and for the hadronic light-by-light scattering (right) contributions to a_μ.

1.1 Hadronic Contributions to the Anomalous Magnetic Moment of the Muon

The main objective of our project is a first-principles determination of the hadronic contributions to the anomalous magnetic moment of the muon $a_\mu \equiv \frac{1}{2}(g-2)_\mu$, the deviation of the gyromagnetic ratio g_μ from the exact value of 2 predicted by the Dirac equation. This quantity exhibits a persistent discrepancy of 3.7 standard deviations between the direct measurement and the consensus value for its newest theoretical prediction based on the SM, published recently by the *Muon g − 2 Theory Initiative* [2]

$$a_\mu = \begin{cases} 116\,592\,089(63) \cdot 10^{-11} & \text{(experiment)} \\ 116\,591\,810(43) \cdot 10^{-11} & \text{(SM prediction)} \end{cases} . \tag{1}$$

While the hadronic contribution to a_μ is small, it dominates the uncertainty of the SM prediction. It is given by a_μ^{hvp}, the contribution from the hadronic vacuum polarisation (HVP), and by a_μ^{hlbl}, the contribution from hadronic light-by-light scattering (HLbL), which are illustrated by the diagrams in Fig. 1. The most accurate prediction of a_μ^{hvp} uses the experimentally determined cross section $e^+e^- \to$ hadrons as input to dispersion theory [3–5]. This theory estimate is subject to experimental uncertainties. Until recently, the hadronic light-by-light scattering contribution has only been determined via model estimates [5–8] with a large and difficult-to-quantify uncertainty.

The calculation is spurred by new experiments (E989 at Fermilab [9, 10] and E34 at J-PARC [11]), which will reduce the error of the direct experimental determination of a_μ by a factor four in the years ahead, requiring highly precise estimates for a_μ^{hvp} and a_μ^{hlbl}.

In Lattice QCD, physical observables of the discretised theory of the strong interaction are calculated by Monte-Carlo integration. The effects of the discretisation and the finite volume have to be controlled by calculations with different lattice spacings and different lattice volumes. Due to computational challenges, most past and current calculations are also performed at unphysical quark masses. Obtaining precise results for a_μ^{hvp} and a_μ^{hlbl} at or close to the physical value of the light (up and

down) and strange quark masses is a crucial step towards a decisive test of the SM with much increased sensitivity, and the main objective of our current calculations on HazelHen at HLRS.

1.2 Hadronic Contributions to Electromagnetic and Electroweak Precision Observables

The fine-structure constant $\alpha = 1/137.035\,999\,139(31)$ [12] is known from low-energy experiments to better than a part per billion. However, its effective value for physics around the Z pole is $\hat{\alpha}^{(5)}(M_Z) = 1/127.955(10)$ [12], a 7 % larger value. The running of α at a given time-like momentum transfer q^2 is described by

$$\alpha(q^2) = \frac{\alpha}{1 - \Delta\alpha(q^2)}, \tag{2}$$

in terms of the $\Delta\alpha(q^2)$ function. While the leptonic contribution to $\Delta\alpha(q^2)$ can be computed in perturbation theory, the estimate of the quark contribution at low energies requires non-perturbative calculations of hadronic physics. Just like discussed for a_μ^{hvp} in the previous section, $\Delta\alpha_{\text{had}}(q^2)$ is usually determined using the total cross-section $e^+e^- \to$ hadrons, which is estimated using a compilation of experimental data. A Lattice QCD determination would provide a valuable independent cross-check.

A key observation that connects the $(g-2)_\mu$ puzzle to $\Delta\alpha_{\text{had}}$ comes from the insight that attributing the tension between experiment and the SM for a_μ to an underestimate of a_μ^{hvp} would lead to a correlated increase of $\Delta\alpha_{\text{had}}$ [13], whose value at the Z-pole enters global SM fits, and in turn would likely create a similar tension there [14]. Lattice QCD determinations consistently estimating both of these quantities are therefore highly desirable.

Another quantity we consider is the electroweak mixing angle or Weinberg angle θ_{W}, which parameterises the mixing between electromagnetic and weak interactions in the SM

$$\sin^2\theta_{\text{W}} = \frac{g'^2}{g^2 + g'^2}, \qquad e = g\sin\theta_{\text{W}} = g'\cos\theta_{\text{W}}, \tag{3}$$

where g and g' are the $\text{SU}(2)_L$ and $\text{U}(1)_Y$ couplings, respectively.

The energy dependence of $\sin^2\theta_{\text{W}}$ can be written as

$$\sin^2\theta_{\text{W}}(q^2) = \sin^2\theta_{\text{W}}\left[1 + \Delta\sin^2\theta_{\text{W}}(q^2)\right], \tag{4}$$

where $\sin^2\theta_{\text{W}}$ is the value in the low-energy limit. The leading hadronic contribution to the running at space-like $Q^2 = -q^2$ is given by [15, 16]

$$\Delta_{\rm had} \sin^2\theta_{\rm W}(-Q^2) = -\frac{e^2}{\sin^2\theta_{\rm W}} \bar{\Pi}^{Z\gamma}(Q^2), \tag{5}$$

where $\bar{\Pi}^{Z\gamma}(Q^2)$ is the HVP mixing of the electromagnetic current and the vector part of the neutral weak current. As in the electromagnetic case, $\bar{\Pi}^{Z\gamma}(Q^2)$ is directly accessible to lattice computations [17–20].

2 Computational Setup

2.1 Observables

Our methodology is based on the expressions derived in the "time-momentum representation" (TMR) [21], *i.e.* a set of integral representations of the observables $a_\mu^{\rm hvp}$, $\Delta\alpha$ and $\Delta_{\rm had}\sin^2\theta_{\rm W}$ in terms of the vector correlator computed in Lattice QCD. The computation of $\bar{\Pi}^{\gamma\gamma}$ and $\bar{\Pi}^{Z\gamma}$ as functions of Q^2 is similar to that of $a_\mu^{\rm hvp}$, the leading-order hadronic vacuum polarization (HVP) contribution to $(g-2)_\mu$,

$$\begin{aligned} a_\mu^{\rm hvp} &= \left(\frac{\alpha}{\pi}\right)^2 \int_0^\infty dx_0\, \widetilde{K}(x_0) G(x_0)\ , \\ \bar{\Pi}(Q^2) &= \int_0^\infty {\rm d}x_0 G(x_0) K(x_0, Q^2)\ , \\ G(x_0) &= -\frac{a^3}{3}\sum_{k=1}^{3}\sum_{\mathbf{x}} \langle J_k(x_0,\mathbf{x}) J_k(0)\rangle\ . \end{aligned} \tag{6}$$

Here $\widetilde{K}(x_0)$ is a known kernel function [21] and $G(x_0)$ denotes the summed correlator involving either the electromagnetic current $J_\mu = \frac{2}{3}\bar{u}\gamma_\mu u - \frac{1}{3}\bar{d}\gamma_\mu d - \frac{1}{3}\bar{s}\gamma_\mu s \cdots$ or the vector part of the neutral weak current J_μ^Z

$$\begin{aligned} J_\mu^Z\big|_{\rm vector} &= J_\mu^{T_3}\big|_{\rm vector} - \sin^2\theta_{\rm W} J_\mu^\gamma, \\ J_\mu^{T_3}\big|_{\rm vector} &= \frac{1}{4}\bar{u}\gamma_\mu u - \frac{1}{4}\bar{d}\gamma_\mu d - \frac{1}{4}\bar{s}\gamma_\mu s + \frac{1}{4}\bar{c}\gamma_\mu c. \end{aligned} \tag{7}$$

To improve statistics, we average over the three spatial directions labelled by k in (6). For $\bar{\Pi}(Q^2)$ we have to integrate over Euclidean time the product of the zero-momentum-projected correlator, $G^{\gamma\gamma}(x_0)$ or $G^{Z\gamma}(x_0)$, times the Q^2-dependent kernel $K(x_0, Q^2) = x_0^2 - (4/Q^2)\sin^2(Qx_0/2)$.

Since there is no firm theoretical guidance for the pion mass dependence of the observables, results computed directly at physical pion mass are of utmost importance. Without their inclusion the commonly used fit *ansätze* are at best phenomenological and can lead to ambiguities of the order of a few percent at the physical

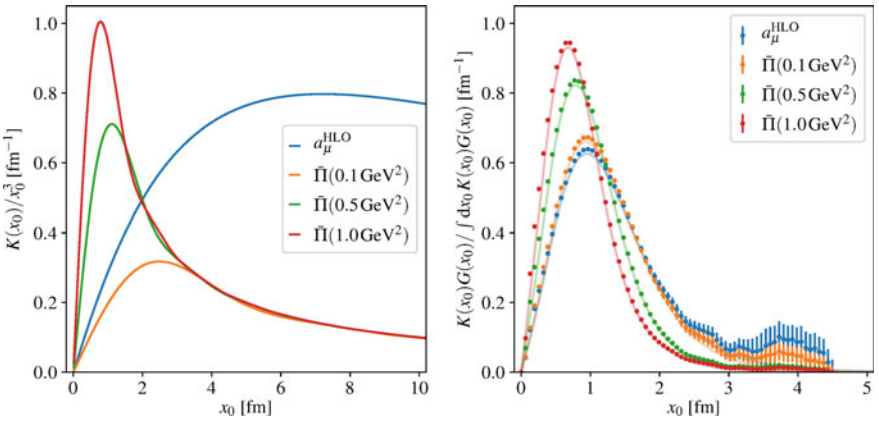


Fig. 2 Left: the kernel $K(x_0, Q^2)$ of the TMR integral divided by x_0^3 for different values of Q^2, compared to the kernel for a_μ^{hvp} [21, 23] (blue line), as a function of time x_0. Right: contribution of $G(x_0)K(x_0, Q^2)$ to the TMR integral normalised to the value of the integral, comparing different kernels $K(x_0)$. The light coloured lines are drawn using a model for the Euclidean-time correlator $G(x_0)$ [21], that is also used for the integral, while the data points with error bars are obtained using actual lattice correlator data at the physical pion mass. (Color figure online)

pion mass [22]. For this reason one of the core components within our project is the generation of a gauge-field ensemble at physical pion mass dubbed E250. The large system size makes E250 an ideal ensemble to be processed on a machine like HazelHen. In addition, previous ensembles with 2+1 flavours of improved Wilson fermions from the *Coordinated Lattice Simulations* (CLS) consortium will be used.

Results from this physical pion mass ensemble for different integration kernels are shown in Fig. 2. A shorter-range kernel puts a larger weight on the correlator at short times. Since the correlator on the lattice is sampled in steps of a spacing a, one needs $Q^2 \ll (\pi/a)^2$ in order to avoid large cut-off effects. On the other hand, a longer-range kernel weights relatively more the long-time behaviour of the correlator, which is noisier and susceptible to finite-volume effects. The latter behaviour can be seen for the kernel for a_μ^{HLO}.

An overview plot of the existing gauge ensembles is shown in Fig. 3. These simulations have been performed using the Hybrid Monte Carlo (HMC) algorithm as implemented in the state-of-the-art `openQCD` code [24]. Progress on the generation of the new gauge ensemble will be described below.

To form the final observables, a second computationally expensive step is the calculation of the quark propagators needed for the solutions ψ of the inhomogeneous Dirac equation for some source vectors η, which are given by

$$\psi = D_{\mathrm{w}}^{-1}\eta. \tag{8}$$

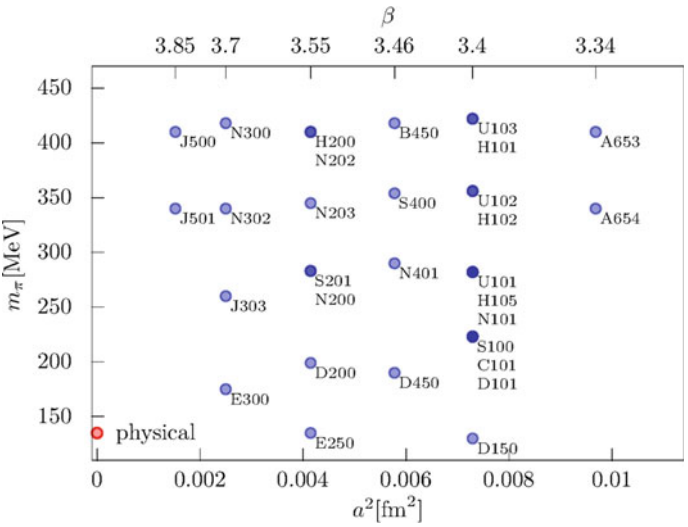


Fig. 3 Landscape of CLS 2+1-flavour ensembles along a trajectory with constant trace of the quark mass matrix $\mathrm{Tr}(M) = \mathrm{const}$. The lattice spacing squared is displayed on the x-axis, while the y-axis shows the pion mass. Multiple ensemble names next to a single dot indicate ensembles with different volumes at the (otherwise) same set of simulation parameters.

Here D_w is the O(a) improved Wilson-Dirac operator. For this step the efficient deflation-accelerated GCR solver from `openQCD` package described in [25, 26] is used. Due to the large system size, the quark propagators on the gauge field ensemble E250 are estimated using stochastic sources, with noise partitioning in spin, colour and time. Each source has support on a single, randomly chosen timeslice. The quark propagators acting on the source vectors are then contracted to form the quark-connected and quark-disconnected correlation functions needed for the calculation of a_μ^hvp and $\bar{\Pi}(Q^2)$. For the simple case of local currents the connected and disconnected Wick contractions are

$$C_{\mu\nu}^{f_1,f_2}(x) = -\left\langle \mathrm{Tr}\left\{ D_{f_1}^{-1}(x,0)\gamma_\mu D_{f_2}^{-1}(0,x)\gamma_\nu \right\} \right\rangle, \tag{9a}$$

$$D_{\mu\nu}^{f_1,f_2}(x) = \left\langle \mathrm{Tr}\left\{ D_{f_1}^{-1}(x,x)\gamma_\mu \right\} \mathrm{Tr}\left\{ D_{f_2}^{-1}(0,0)\gamma_\nu \right\} \right\rangle, \tag{9b}$$

where indices f_1 and f_2 denote quark flavours. These flavours are labelled explicitly with index ℓ (for up and down), s (for strange), and c (for charm) below. With these contractions, the flavour SU(3) contributions are defined as

$$G_{\mu\nu}^{33}(x) = \frac{1}{2} C_{\mu\nu}^{\ell,\ell}(x), \tag{10a}$$

$$G^{88}_{\mu\nu}(x) = \frac{1}{6}\left[C^{\ell,\ell}_{\mu\nu}(x) + 2C^{s,s}_{\mu\nu}(x) + 2D^{\ell-s,\ell-s}_{\mu\nu}(x)\right], \tag{10b}$$

$$G^{08}_{\mu\nu}(x) = \frac{1}{2\sqrt{3}}\left[C^{\ell,\ell}_{\mu\nu}(x) - C^{s,s}_{\mu\nu}(x) + D^{2\ell+s,\ell-s}_{\mu\nu}(x)\right], \tag{10c}$$

The $\gamma\gamma$ bare correlators relevant for $a_\mu^{\rm hvp}$ and $\Delta\alpha$, and the $Z\gamma$ bare correlators relevant for $\theta_{\rm W}$ are then given by

$$G^{\gamma\gamma}_{\mu\nu}(x) = G^{33}_{\mu\nu}(x) + \frac{1}{3}G^{88}_{\mu\nu}(x) + \frac{4}{9}C^{c,c}_{\mu\nu}(x), \tag{11a}$$

$$G^{Z\gamma}_{\mu\nu}(x) = \left(\frac{1}{2} - \sin^2\theta_{\rm W}\right)G^{\gamma\gamma}_{\mu\nu}(x) - \frac{1}{6\sqrt{3}}G^{08}_{\mu\nu}(x) - \frac{1}{18}C^{c,c}_{\mu\nu}(x). \tag{11b}$$

Throughout the calculation, errors are estimated using the jackknife procedure with blocking in order to take into account auto-correlation effects.

2.2 Code Performance and Workflow

The main computational task of the project is the generation of an ensemble of gauge configurations with physical pion and kaon masses, needed to achieve the required precision for the observables described in the previous section. For this task we use version 1.6 of the `openQCD` code [24] developed by Martin Lüscher and project contributor Stefan Schaefer, which is publicly available under the GPL license. For the gauge field generation runs on HazelHen at HLRS we used two setups:

A Local lattice volume of size $12 \times 8 \times 16 \times 16$ per MPI rank with 6912 MPI ranks on 288 nodes
B Local lattice volume of size $12 \times 8 \times 8 \times 16$ per MPI rank with 13824 MPI ranks on 576 nodes

In addition to the gauge field generation, we also performed quark propagator runs on HazelHen with Setup C below:

C Local lattice volume of size $16 \times 16 \times 16 \times 8$ per MPI rank with 5184 MPI ranks on 216 nodes.

In each case a hypercube of $2 \times 3 \times 2 \times 2$ processes was grouped onto a single node, to minimise off-node communication. This process setup was also determined to be optimal by the Cray profiler. Further details on these runs including some of the obstacles encountered can be found in last year's report.

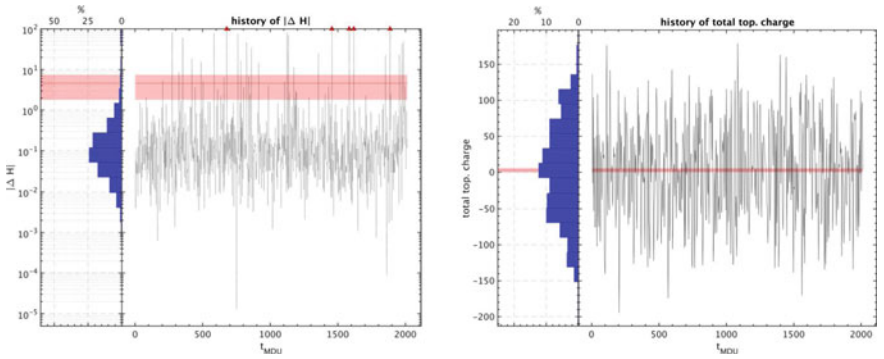


Fig. 4 Monte-Carlo histories of the Hamiltonian deficit ΔH (left) and the total topological charge (right) from the current E250 run performed on HazelHen.

2.3 Status of the Gauge Field Generation

We now report on the status of gauge field generation for ensemble E250 with physical pion mass. The status of these simulations has previously been described in [27]. Until the end of the previous allocation period 503 gauge field configurations corresponding to 2012 molecular dynamics units have been generated on HazelHen. The calculation of observables was performed in parallel on HazelHen and on compute clusters at JGU Mainz. Further gauge field generation runs are planned as part of a new allocation on Hawk.[1]

In Fig. 4 we show two quantities that characterise the efficiency of the algorithm: the left pane shows the Monte Carlo history of the Hamiltonian deficit ΔH; the absence of any severe spikes demonstrates that the algorithmic setup is very stable, resulting in a high acceptance rate $(87.7 \pm 1.2)\%$ even for this demanding simulation at the physical pion mass. The right pane of Fig. 4 shows the Monte Carlo history of the topological charge: despite the fact that the lattice spacing is small, one clearly sees a high tunnelling frequency, which provides evidence that the algorithm correctly samples different topological sectors, if a chain of sufficient length is simulated.

3 Hadronic Contributions the Anomalous Magnetic Moment of the Muon

In this section we briefly present the results for the hadronic vacuum polarisation contribution to a_μ^{hvp} published in [28] and previously summarised in the last report. We obtained the result

[1] As of writing this report these calculations have started but are not part of the reporting period covered here.

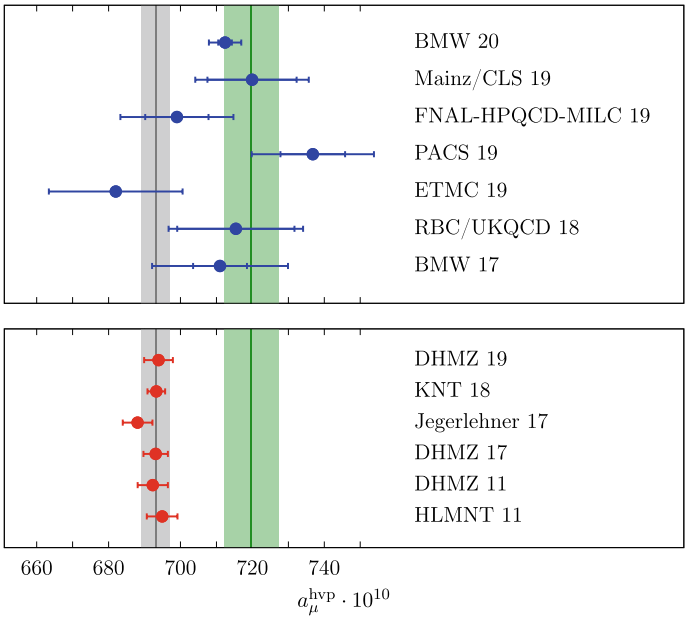


Fig. 5 Overview of results for a_μ^{hvp} from Lattice QCD (top) and from dispersive methods (bottom) compared to a "no new physics" scenario (green band). The SM prediction [2] is the grey band. Our published result [28] is labelled "Mainz/CLS 19".

$$a_\mu^{\mathrm{hvp}} = (720.0 \pm 12.4_{\mathrm{stat}} \pm 9.9_{\mathrm{syst}}) \cdot 10^{-10}, \tag{12}$$

where the first error is statistical, and the second is an estimate of the total systematic uncertainty, which among other things accounts for the fact that the corrections due to isospin breaking have not yet been included in the central value. We thus find that the current overall error of our determination is 2.2%.

Figure 5 shows an overview of recent results for a_μ^{hvp} from Lattice QCD and from dispersive methods. The blue data points in the top pane show lattice results, while the bottom pane shows the results from dispersive analyses. The green band is the "no new physics" scenario in which the discrepancy between the SM estimate of a_μ and the experimental observation is attributed purely to a_μ^{hvp}. The grey band is the SM estimate from [2], which is largely compatible with the single results using dispersive methods and experimental data. Using our ongoing calculations based on ensemble E250 we are currently improving the overall accuracy of our result.

The central value of the theoretical prediction for a_μ^{hlbl} is mostly based on hadronic model calculations which offer poor control over the associated uncertainty. Although it is a sub-leading hadronic contribution it has a large associated uncertainty, and so this quantity is becoming more and more relevant as experimental precision increases. The discrepancy between theory and experiment appears to persist, so it is vital to accurately compute the HLbL contribution in a systematically improvable way.

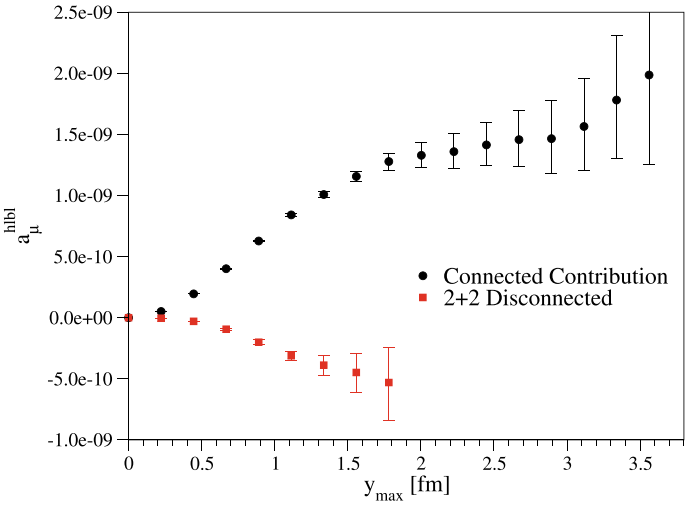


Fig. 6 The accumulated connected and leading disconnected (2+2) contributions to a_μ^{hlbl} as a function of the upper integration bound y_{max}.

For the HLbL calculation there are 5 topologies of diagrams that contribute. In order of expected significance they are the connected and various types of disconnected diagrams, of which the (2+2) topology is expected to be the most important one. It is important to realise that accurate determinations of the individual connected and the leading disconnected diagram are cutting-edge calculations. The (2+2) contribution is important to determine with good accuracy as it appears to be comparable in magnitude to that of the connected piece but with opposite sign [29]; as the (2+2) is a disconnected contribution it will be statistically much noisier than the connected diagram.

In Fig. 6 we show some preliminary results from the ensemble D200 for the quantity,

$$a_\mu^{\text{hlbl}} = \frac{me^6}{3} 2\pi^2 \int_0^{y_{\text{max}}} |y|^3 \, \mathrm{d}|y| \int d^4x \; \bar{\mathcal{L}}_{[\rho,\sigma];\mu\nu\lambda}(x, y) \; i\Pi_{\rho;\mu\nu\lambda\sigma}(x, y), \qquad (13)$$

which allows us to infer where the integral has saturated as a function of the upper integration bound y_{max}. Π in Eq. 13 denotes the Wick-contractions for different topologies of four-point functions, two of which are most important in the hadronic-light-by-light contribution to the $g-2$ of the muon, the connected and the $2+2$ disconnected. The sum of the two results gives us a measurement (neglecting subleading disconnected topologies) of the full contribution. Much like the conclusions drawn from phenomenological estimates and other lattice studies, the light-by-light contribution to $g-2$ is far too small to explain the discrepancy between theory and experiment.

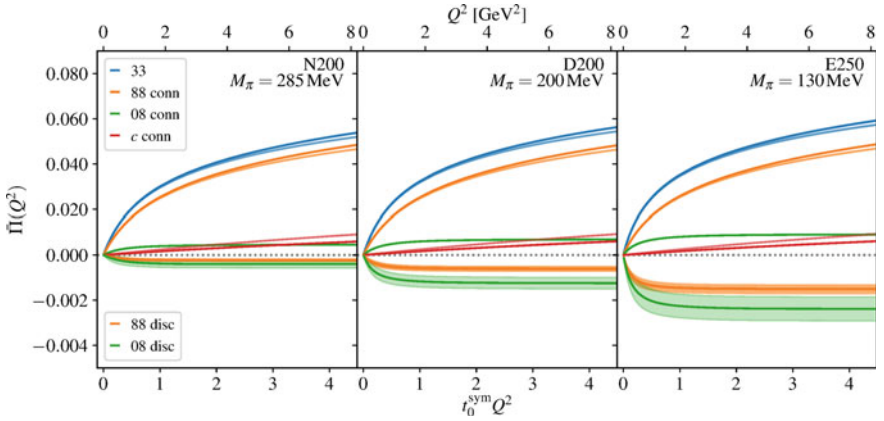


Fig. 7 Running with energy Q^2 of different contributions to $\bar{\Pi}(Q^2)$ on three different ensembles at $a \approx 0.064$ fm. The conserved-local discretisation is shown and, when available, the local, local discretisation in a lighter colour shade. The negative side of the vertical axis of the plot is inflated by a factor 10 with respect to the positive side.

For the calculation of the connected contribution we have used 250 configurations of the ensemble D200, with 8 different orientations of the direction $|y|$ in order to further enhance statistical precision. For the $2 + 2$ disconnected contribution we have used 82 configurations each of which requires hundreds of propagator solves. The results presented here for D200 are very consistent with a similar, coarser lattice-spacing, ensemble C101 suggesting that the extrapolation to the continuum limit will be under control. Having results for the ensemble D200 allows us to perform a robust chiral and continuum extrapolation of $a_\mu^{\rm hlbl}$ in conjunction with other measurements performed elsewhere.[2]

4 Hadronic Running of Electromagnetic and Electroweak Couplings

Figure 7 shows the running of different contributions to $\bar{\Pi}(Q^2)$, defined through the correlators in Eq. (11), as a function of Q^2 on three different gauge field ensembles at the same lattice spacing with increasingly lighter pions. The rightmost pane shows the running for the E250 ensemble with physical pion masses generated as part of this project. The negative disconnected contributions are shown enlarged by a factor 10 and show a significant increase towards lighter pion masses.

Figure 8 shows the extrapolation of the isovector and isoscalar contributions to the running of α, $\bar{\Pi}^{33}(Q^2)$ and $\bar{\Pi}^{88}_{\rm conn}(Q^2)$ to the continuum limit ($a = 0$) and to physical pion and kaon masses. From the results at the physical point we obtain preliminary

[2] For a recent preprint on $a_\mu^{\rm hlbl}$ at heavier pion masses see [30].

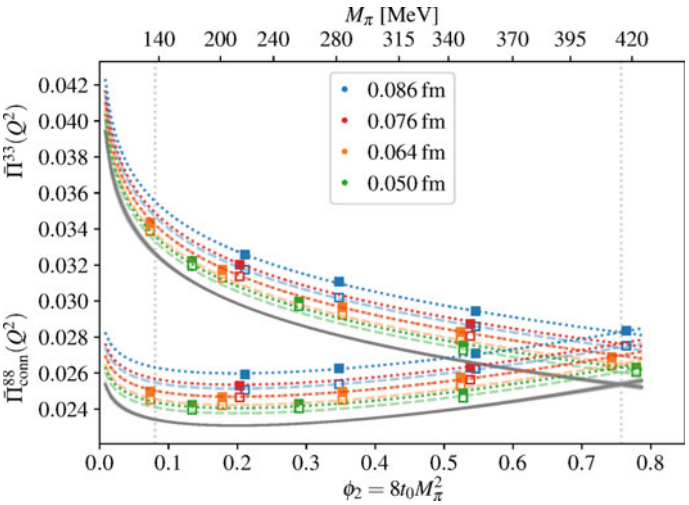


Fig. 8 Combined extrapolation of $\bar{\Pi}^{33}(Q^2)$ and $\bar{\Pi}^{88}_{\rm conn}(Q^2)$ at $Q^2 = 1\,\text{GeV}^2$ to the physical point. Filled symbols denote the conserved, local discretisation, while open symbols denote the local, local one.

estimates for the leading hadronic connected contribution from QCD to the running of α and $\sin^2\theta_{\rm W}$ at $Q^2 = 1\,\text{GeV}^2$

$$\begin{aligned} \Delta\alpha_{\rm had}(-1\text{GeV}^2) &= 0.003\,708(18)\,, \\ \Delta_{\rm had}\sin^2\theta_{\rm W}(-1\text{GeV}^2) &= -0.003\,764(17)\,. \end{aligned} \tag{14}$$

These results represent fully non-perturbative predictions from QCD and can be compared with corresponding phenomenological estimates based on experimental data.

5 Summary and Outlook

Precision observables such as the muon anomalous magnetic moment, the electromagnetic coupling and the electroweak mixing angle play a central role in precision tests of the SM. In order to match the accuracy of direct measurements, it is mandatory to have full control over the effects from the strong interaction. Lattice QCD is a versatile and mature method that allows for the calculation of hadronic contributions to these precision observables from first principles, using Monte Carlo integration. A crucial ingredient is the ability to perform simulations at the physical value of the pion mass: due to the associated large system size, the problem is ideally suited to run on massively parallel systems such as Hazel Hen. Nevertheless, despite the availability of efficient simulation algorithms, it is necessary to control fluctuations

in the spectrum of the Wilson-Dirac operator [31] which is our chosen discretisation of the quark action.

Our results for the hadronic contributions to the muon anomalous magnetic moment are not yet precise enough to discriminate between the current tension of 3.7 standard deviations and the "no new physics" scenario. We are currently extending our calculations to include more statistics, improve the estimation of quark-disconnected diagrams, perform an in-depth study of the long-distance contribution to the convolution integrals, as well as include the effects of strong and electromagnetic isospin breaking [32–34].

Acknowledgements This work is partly supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) grant HI 2048/1-1 and by the DFG-funded Collaborative Research Centre SFB 1044 *The low-energy frontier of the Standard Model*. The Mainz $(g-2)_\mu$ project is also supported by the Cluster of Excellence *Precision Physics, Fundamental Interactions, and Structure of Matter* (PRISMA+ EXC 2118/1) funded by the DFG within the German Excellence Strategy (Project ID 39083149). This work is supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme through grant agreement 771971-SIMDAMA. Calculations for this project were partly performed on the HPC clusters "Clover" and "HIMster II" at the Helmholtz-Institut Mainz and "MOGON II" at JGU Mainz. The authors also gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this project by providing computing time on the GCS Supercomputer HazelHen at Höchstleistungsrechenzentrum Stuttgart (www.hlrs.de) under project GCS-HQCD. We are grateful to our colleagues in the CLS initiative for sharing ensembles.

References

1. R.K. Ellis et al., arXiv:1910.11775
2. T. Aoyama et al., Phys. Rept. **887**, 1–166 (2020). https://doi.org/10.1016/j.physrep.2020.07.006
3. M. Davier, A. Hoecker, B. Malaescu, Z. Zhang, Eur. Phys. J. **C71**, 1515 (2011). https://doi.org/10.1140/epjc/s10052-012-1874-8. https://doi.org/10.1140/epjc/s10052-010-1515-z. [Erratum: Eur. Phys. J. **C72**, 1874 (2012)]
4. K. Hagiwara, R. Liao, A.D. Martin, D. Nomura, T. Teubner, J. Phys. **G38**, 085003 (2011). https://doi.org/10.1088/0954-3899/38/8/085003
5. T. Blum et al., arXiv:1311.2198
6. F. Jegerlehner, A. Nyffeler, Phys. Rept. **477**, 1 (2009). https://doi.org/10.1016/j.physrep.2009.04.003
7. J. Prades, E. de Rafael, A. Vainshtein, Adv. Ser. Direct. High Energy Phys. **20**, 303 (2009). https://doi.org/10.1142/9789814271844_0009
8. J. Bijnens, EPJ Web Conf. **118**, 01002 (2016). https://doi.org/10.1051/epjconf/201611801002
9. J. Kaspar, Nucl. Part. Phys. Proc. **260**, 243 (2015). https://doi.org/10.1016/j.nuclphysbps.2015.02.051
10. M. Fertl, Hyperfine Interact. **237**(1), 94 (2016). https://doi.org/10.1007/s10751-016-1304-7
11. M. Otani, JPS Conf. Proc. **8**, 025010 (2015). https://doi.org/10.7566/JPSCP.8.025010
12. M. Tanabashi et al., Phys. Rev. D **98**(3), 030001 (2018). https://doi.org/10.1103/PhysRevD.98.030001
13. M. Passera, W. Marciano, A. Sirlin, Phys. Rev. D **78**, 013009 (2008). https://doi.org/10.1103/PhysRevD.78.013009
14. A. Keshavarzi, W.J. Marciano, M. Passera, A. Sirlin, arXiv:2006.12666

15. F. Jegerlehner, Z. Phys. C **32**, 195 (1986). https://doi.org/10.1007/BF01552495
16. F. Jegerlehner, Nuovo Cim. C **034S1**, 31 (2011). https://doi.org/10.1393/ncc/i2011-11011-0
17. F. Burger, K. Jansen, M. Petschlies, G. Pientka, JHEP **11**, 215 (2015). https://doi.org/10.1007/JHEP11(2015)215
18. A. Francis et al., PoS LATTICE2015, 110 (2015)
19. V. Gülpers, H. Meyer, G. von Hippel, H. Wittig, PoS LATTICE2015, 263 (2016). https://doi.org/10.22323/1.251.0263
20. M. Cè, A. Gérardin, K. Ottnad, H.B. Meyer, PoS LATTICE2018, 137 (2018). https://doi.org/10.22323/1.334.0137
21. D. Bernecker, H.B. Meyer, Eur. Phys. J. A **47**, 148 (2011). https://doi.org/10.1140/epja/i2011-11148-6
22. M. Golterman, K. Maltman, S. Peris, Phys. Rev. **D95**(7), 074509 (2017). https://doi.org/10.1103/PhysRevD.95.074509
23. M. Della Morte et al., JHEP **10**, 020 (2017). DOI https://doi.org/10.1007/JHEP10(2017)020
24. M. Lüscher, S. Schaefer, Comput. Phys. Commun. **184**, 519 (2013). https://doi.org/10.1016/j.cpc.2012.10.003
25. M. Lüscher, JHEP **07**, 081 (2007). https://doi.org/10.1088/1126-6708/2007/07/081
26. M. Lüscher, JHEP **0712**, 011 (2007). https://doi.org/10.1088/1126-6708/2007/12/011
27. D. Mohler, S. Schaefer, J. Simeth, EPJ Web Conf. **175**, 02010 (2018). https://doi.org/10.1051/epjconf/201817502010
28. A. Gérardin, et al.: Phys. Rev. D **100**(1), 014510 (2019). https://doi.org/10.1103/PhysRevD.100.014510
29. T. Blum et al., Phys. Rev. Lett. **118**(2), 022005 (2017). https://doi.org/10.1103/PhysRevLett.118.022005
30. E.H. Chao, A. Gérardin, J.R. Green, R.J. Hudspith, H.B. Meyer, arXiv:2006.16224
31. D. Mohler, S. Schaefer, Phys. Rev. D **102**, 074506 (2020). https://doi.org/10.1103/PhysRevD.102.074506
32. A. Risch, H. Wittig, EPJ Web Conf. **175**, 14019 (2018). https://doi.org/10.1051/epjconf/201817514019
33. A. Risch, H. Wittig, PoS LATTICE2018, 059 (2018). https://doi.org/10.22323/1.334.0059
34. A. Risch, H. Wittig, arXiv:1911.04230

MCTDH-X: The Multiconfigurational Time-Dependent Hartree Method for Indistinguishable Particles High-Performance Computation Project

A. U. J. Lode, O. E. Alon, M. A. Bastarrachea-Magnani, A. Bhowmik, A. Buchleitner, L. S. Cederbaum, R. Chitra, E. Fasshauer, L. de Forges de Parny, S. K. Haldar, C. Lévêque, R. Lin, L. B. Madsen, P. Molignini, L. Papariello, F. Schäfer, A. I. Streltsov, M. C. Tsatsos, and S. E. Weiner

Abstract This report introduces the multiconfigurational time-dependent Hartree method for indistinguishable particles (MCTDH-X) high performance computation project and its recent research results. We solved the many-particle Schrödinger equation for time-dependent and time-independent systems using the software implementations of theories in the MCTDH-X family on high-performance computation facilities. Going beyond the commonly applied semi-classical and mean-field pictures, we unveil fascinating and fundamental many-body physics in the correlated electron dynamics within the photoionization of neon, ultracold bosons' hierarchical superfluidity in a cavity, as well as the dynamics of fragmentation, entropy, angular momentum, correlations, and fluctuations of interacting bosons and fermions in one- and two-dimensional double-well potentials. Our present report illustrates how the computational resources at the HLRS for our MCTDH-X applications enabled and boosted our scientific research productivity in the field of many-body physics.

A. U. J. Lode (✉)
Physikalisches Institut, Albert-Ludwigs-Universität Freiburg, Hermann-Herder-Str. 3, 79104 Freiburg, Germany
e-mail: auj.lode@gmail.com

O. E. Alon · A. Bhowmik · S. K. Haldar
Department of Mathematics, University of Haifa, 3498838 Haifa, Israel
e-mail: ofir@research.haifa.ac.il

Haifa Research Center for Theoretical Physics and Astrophysics, University of Haifa, 3498838 Haifa, Israel

M. A. Bastarrachea-Magnani · A. Buchleitner · L. de Forges de Parny · F. Schäfer
Physikalisches Institut, Albert-Ludwigs-Universität Freiburg, Hermann-Herder-Straße 3, 79104 Freiburg, Germany

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_2

1 Introduction

The multiconfigurational time-dependent Hartree method for indistinguishable particles [1–11, 128, 129] (MCTDH-X) enjoys a growing amount of scientific interest and activity greatly spurred by the availability of high-performance computation facilities like the Hazel Hen and the just started Hawk clusters at the HLRS in Stuttgart.

Here, we report our research activities solving the many-particle Schrödinger equation with the software implementations of methods in the family of MCTDH-X theories: the MCTDHB, MCTDHB- LAB, and MCTDH- X software packages, see [12–14].

With our computations we were able to reveal and describe novel and exciting fundamental properties of the quantum many-body states of ultracold atoms that we published in Refs. [15–20].

The research that we document in the present report is adding to our long and prosperous series of results enabled and corroborated via our access to the HLRS Stuttgart platforms for high-performance computation [21–27].

The structure of our report is as follows: in Sect. 2, we introduce the MCTDH-X method, in Sect. 3 we collect the quantities of interest that we use to investigate the many-body physics in Refs. [15–20], in Sect. 4 we analyze the photoionization of neon hit by a laser pulse, in Sect. 5 we study the quantum phases of laser-pumped bosons immersed in a high-finesse optical cavity, in Sect. 6 we compare the dynamical behavior of bosons and fermions in double-well potentials, in Sect. 7 we consider the dynamical and spectral features and entropy of bosonic particles in double wells, in Sect. 8 we consider the universality of fragmentation and resonant tunneling in an asymmetric bosonic Josephson junction, in Sect. 9 we consider the tunneling dynamics in a two-dimensional bosonic Josephson junction and the effects of the transverse direction on it, and in Sect. 10 we conclude with a summary and outlook.

M. A. Bastarrachea-Magnani
Department of Physics and Astronomy, Aarhus University, Ny Munkegade, 8000 Aarhus C, Denmark

A. Buchleitner
Freiburg Institute for Advanced Studies (FRIAS), Albert-Ludwigs-Universität Freiburg, Albertstr. 19, 79104 Freiburg, Germany

L. S. Cederbaum · A. I. Streltsov
Theoretische Chemie, Physikalisch-Chemisches Institut, Universität Heidelberg, Im Neuenheimer Feld 229, 69120 Heidelberg, Germany
e-mail: lorenz.cederbaum@pci.uni-heidelberg.de

R. Chitra · R. Lin · P. Molignini · L. Papariello
Institute for Theoretical Physics, ETH Zürich, 8093 Zürich, Switzerland

2 MCTDH-X: The Multiconfigurational Time-Dependent Hartree Method for Indistinguishable Particles

We now discuss the multiconfigurational time-dependent Hartree method for indistinguishable particles for full [1–11] and restricted configuration spaces [28–32].

For the archetypical MCTDH-X with a complete configuration space, so-called full configuration interaction (FCI), the wavefunction is represented as a time-dependent superposition of all possible configurations of N particles in M time-dependent orbitals:

$$|\Psi^{FCI}\rangle = \sum_{\mathbf{n}} C_{\mathbf{n}}(t)|\mathbf{n};t\rangle; \quad \mathbf{n} = (n_1, ..., n_M)^T;$$
$$|\mathbf{n};t\rangle = \mathcal{N}\prod_{i=1}^{M}\left[\hat{b}_i^\dagger(t)\right]^{n_i}|vac\rangle. \tag{1}$$

Here, $n_i, i = 1, ..., M$ are M integer occupation numbers, the total particle number is $\sum_i n_i = N$, and the normalization $\mathcal{N}$ is $\frac{1}{\sqrt{N!}}$ ($\frac{1}{\sqrt{\prod_{i=1}^{M} n_i!}}$) for fermions (bosons). The annihilation operator $\hat{b}_j^\dagger$ are connected to the time-dependent orbitals $\Phi_j(\mathbf{r},t)$ as follows:

$$\Phi_j(\mathbf{r},t) = \langle\mathbf{r}|\hat{b}_j^\dagger(t)|vac\rangle. \tag{2}$$

See Fig. 1 for a pictorial representation of $|\Psi^{FCI}\rangle$.

In this report we use $\mathbf{r}$ to collect the degrees of freedom (spin and space) of the orbitals. The coefficients can be computed from the wavefunction by projecting it onto a specific configuration $|\mathbf{n};t\rangle$:

$$C_{\mathbf{n}}(t) = \langle\mathbf{n};t|\Psi^{FCI}\rangle. \tag{3}$$

P. Molignini
Clarendon Laboratory, Department of Physics, University of Oxford, Oxford OX1 3PU, UK

L. Papariello
Research Studio Data Science, RSA FG, 1090 Vienna, Austria

F. Schäfer
Department of Physics, University of Basel, Klingelbergstrasse 82, 4056 Basel, Switzerland

M. C. Tsatsos
São Carlos Institute of Physics, University of São Paulo, P.O. Box 369, São Carlos, São Paulo 13560-970, Brazil

S. E. Weiner
Department of Physics, UC Berkeley, Berkeley, CA, USA

E. Fasshauer
Department of Physics and Astronomy, Ny Munkegade 120, building 1520, 630, 8000 Aarhus C, Denmark

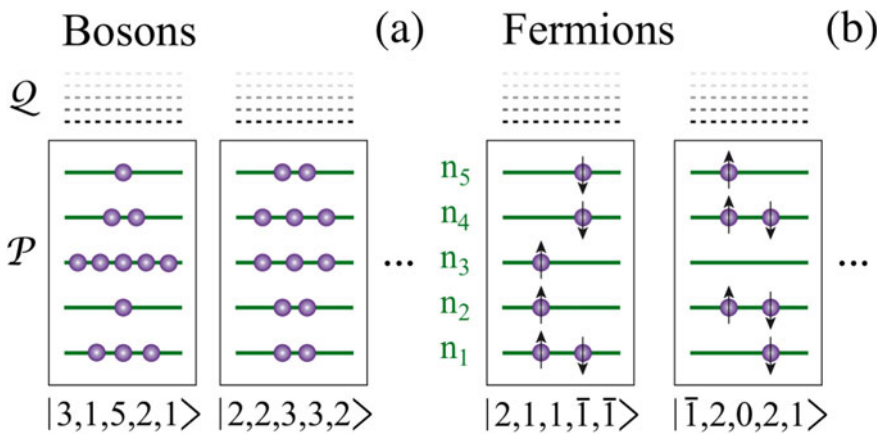


Fig. 1 Sketch of the FCI configuration space of MCTDH-X for bosons (a) and fermions (b). The depicted configurations are given as occupation number vectors $|n_1, n_2, ...\rangle$. Here, a $\bar{1}$ indicates spin-down and 1 indicates spin-up for fermions in (b). The partitioning of Hilbert space into the space $\mathcal{P}$ of occupied orbitals and its complement $\mathcal{Q}$ of unoccupied orbitals is indicated on the left. Figure reprinted from [15].

The number of coefficients in the FCI space is $\binom{M}{N}$ for fermions and $\binom{N+M-1}{N}$ for bosons. For large particle numbers N and/or a large number of orbitals M, the number of coefficients can become prohibitively large for practical numerical computations. In order to save numerical effort by reducing the number of coefficients, the restricted active space (RAS) approach from quantum chemistry [33] can be used to provide a recipe for selecting the configurations in the considered Hilbert space $\mathcal{V}$. The resulting RAS-MCTDH-X ansatz reads:

$$|\Psi^{RAS}\rangle = \sum_{\mathbf{n}\in\mathcal{V}} C_{\mathbf{n}}(t)|\mathbf{n}, t\rangle. \tag{4}$$

For a pictorial representation of the construction of the restricted Hilbert space $\mathcal{V}$ and $|\Psi^{RAS}\rangle$, see Fig. 2.

L. de Forges de Parny
ACRI-ST, 260 route du Pin Montard, 06904 Sophia Antipolis Cedex, France

S. K. Haldar
Department of Physics, SRM University Delhi-NCR, Plot No. 39 Rajiv Gandhi education city, Sonipat 131029, India

C. Lévêque
Vienna Center for Quantum Science and Technology, Atominstitut, TU Wien, Stadionallee 2, 1020 Vienna, Austria

Wolfgang Pauli Institute c/o Faculty of Mathematics, University of Vienna, Oskar-Morgenstern Platz 1, 1090 Vienna, Austria
L. B. Madsen
Department of Physics and Astronomy, Aarhus University, 8000 Aarhus C, Denmark

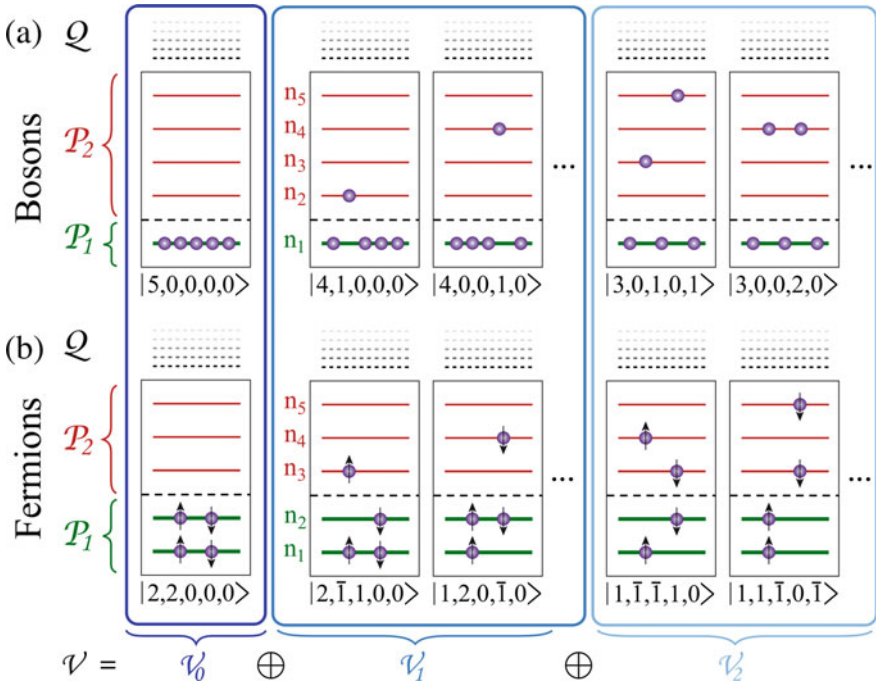


Fig. 2 Sketch of the construction of the restricted active space MCTDH-X wavefunction $|\Psi^{RAS}\rangle$ for bosons (a) and fermions (b). The $\mathcal{P}$-space of occupied orbitals is partitioned into a $\mathcal{P}_1$-subspace where all possible configurations are taken into account and a $\mathcal{P}_2$ subspace, for which the maximal number of particles is restricted. The total restricted Hilbert space $\mathcal{V}$ is then a direct sum of the spaces $\mathcal{V}_i = \mathcal{P}_1(N-i) \oplus \mathcal{P}_2(i)$ with at most i particles in $\mathcal{P}_2$: $\mathcal{V} = \mathcal{V}_0 \oplus \mathcal{V}_1 \oplus \mathcal{V}_2 \oplus \cdots \oplus \mathcal{V}_i$. Figure reprinted from [15].

The above two ansatzes, Eqs. (1) and (4), yield different equations of motion when combined with the time-dependent variational principle [34–36]: Eq. (1) yields the standard MCTHD-X equations [1–3] and Eq. (4) yields the RAS-MCTDH-X equations [28–32]. Both methods and further details on their generalizations [37–40], applications [41–66], and verification with experimental results [67, 68] are reviewed in Ref. [15].

3 Quantities of Interest

We discuss the quantities that are used in this report to analyze the many-body physics in the following. In order to quantify the degree of condensation of a given bosonic many-body state [69], we resort to the eigenvalues and eigenfunctions of the one-body reduced density matrix (1-RDM), ρ_i and φ_i, respectively:

$$\rho^{(1)}(\mathbf{r},\mathbf{r}';t) = N\mathrm{Tr}_{\mathbf{r}_2,\ldots,\mathbf{r}_N}\left[|\Psi\rangle\langle\Psi|\right] = \sum_i \rho_i(t)\varphi_i^*(\mathbf{r}';t)\varphi_i(\mathbf{r};t). \tag{5}$$

A bosonic state is condensed when a single ρ_i is macroscopic [69]. The state is fragmented if multiple ρ_i are macroscopic [1, 70–82].

The Glauber correlation functions yield position-resolved information about the coherence of the many-body state [83, 84]. We will analyze the first-order correlation function that can be obtained from the 1-RDM:

$$g^{(1)}(\mathbf{r},\mathbf{r}';t) = \frac{\rho^{(1)}(\mathbf{r},\mathbf{r}';t)}{\sqrt{\rho^{(1)}(\mathbf{r},\mathbf{r};t)\rho^{(1)}(\mathbf{r}',\mathbf{r}';t)}}. \tag{6}$$

In the illustrative examples to be discussed below, we use x instead of $\mathbf{r}$ to denote the spatial coordinate whenever we study one-dimensional problems. The momentum-space representations of the 1-RDM, $\tilde{\rho}^{(1)}$, and the first-order Glauber correlation function, $\tilde{g}^{(1)}$, are obtained by replacing the state $\Psi(\mathbf{r}_1,\ldots,\mathbf{r}_N;t)$ by its momentum space representation $\tilde{\Psi}(\mathbf{k}_1,\ldots,\mathbf{k}_N;t)$ in Eqs. (5) and (6), respectively.

The eigenvalues of the 1-RDM can furthermore be used to characterize different quantum phases in optical lattices. For this purpose, we define the occupation order parameter Δ (see [85]):

$$\Delta = \sum_{i=1}^{M}\left(\frac{\rho_i}{N}\right)^2. \tag{7}$$

For a condensed state, the single eigenvalue ρ_1 contributes and Δ is close to unity. For a Mott-insulator, the 1-RDM has as many (quasi-)degenerate eigenvalues as there are sites in the lattice [86]. For an S-site N-particle Mott-insulator $\Delta = N/S$ [87]. In the crystal phase of dipolar bosons, each particle sits in a separate natural orbital and, thus, $\Delta = 1/N$, see [85].

Moreover, the eigenvalues of the 1-RDM can provide a measure for the entropy and, therewith, the entanglement embedded in bosonic many-body states [18, 45, 86, 88, 89]:

$$S(t) = -\sum_{i=1}^{M}\frac{\rho_i(t)}{N}\ln\left(\frac{\rho_i(t)}{N}\right). \tag{8}$$

Whenever $S(t)$ is close to 0, the many-body state is factorizable, a mean-field description is close to exact, and entanglement is absent. When $S(t)$ becomes large, the many-body state is non-factorizable, mean-field description is not applicable [45], and entanglement is present.

We will analyze the many-body state by simulating the conventional measurement technique used for cold-atom systems: absorption or single-shot images. A single-shot image $\mathbf{s} = (s_1,\ldots,s_N)^T$ is a sample distributed according to the N-body probability density $P(\mathbf{r}_1,\ldots,\mathbf{r}_N;t) = |\Psi|^2 \equiv \rho^{(N)}(\mathbf{r}_1,\ldots,\mathbf{r}_N;t)$,

$$\mathbf{s} \sim P(\mathbf{r}_1,\ldots,\mathbf{r}_N;t). \tag{9}$$

We use MCTDH-X to simulate these single-shot images [17, 85, 90, 91] and extract the position-dependent density fluctuations:

$$\mathcal{V}(\mathbf{r}) = \frac{1}{N_{\text{shot}}} \sum_{i=1}^{N_{\text{shot}}} \mathcal{B}_i^2(\mathbf{r}) - \left[\frac{1}{N_{\text{shot}}} \sum_{i=1}^{N_{\text{shot}}} \mathcal{B}_i(\mathbf{r}) \right]^2 . \tag{10}$$

Here, $\mathcal{B}_i(\mathbf{r})$ is the value of the i-th simulated single-shot image at position $\mathbf{r}$ and N_{shot} is the number of considered images.

To analyze photoionization of many-electron atoms, the cross section σ_1 can be obtained from a solution to the time-dependent many-electron problem with an n_p-cycle laser-pulse of peak intensity I_0 and angular frequency ω by extracting the ionization probability P_1 (see Refs. [92, 93]) as follows:

$$\sigma_1(Mb) = 1.032 \times 10^{14} \omega^2 \frac{P_1}{n_p I_0}. \tag{11}$$

The variance of an observable $\hat{A}$ for a quantum system described by the state $|\Psi(t)\rangle$ is associated with the resolution with which $\hat{A}$ can be measured on $|\Psi(t)\rangle$. The variance requires the evaluation of the expectation values of $\hat{A}$ and the square of $\hat{A}$. We refer for simplicity to structureless bosons, with a straightforward generalization to indistinguishable particles with spin. Whereas $\hat{A} = \sum_{j=1}^{N} \hat{a}(\mathbf{r}_j)$ consists of one-body operators only, its square $\hat{A}^2 = \sum_{j=1}^{N} \hat{a}^2(\mathbf{r}_j) + \sum_{j<k}^{N} 2\hat{a}(\mathbf{r}_j)\hat{a}(\mathbf{r}_k)$ is a sum of one-body and two-body operators. All in all, the variance can be expressed as [19]

$$\begin{aligned}
&\frac{1}{N} \Delta^2_{\hat{A}}(t) = \frac{1}{N} [\langle \Psi(t)|\hat{A}^2|\Psi(t)\rangle - \langle \Psi(t)|\hat{A}|\Psi(t)\rangle^2] = \\
&= \frac{1}{N} \Bigg\{ \sum_j \rho_j(t) \int d\mathbf{r} \varphi_j^*(\mathbf{r};t) \hat{a}^2(\mathbf{r}) \varphi_j(\mathbf{r};t) - \\
&- \left[\sum_j \rho_j(t) \int d\mathbf{r} \varphi_j^*(\mathbf{r};t) \hat{a}(\mathbf{r}) \varphi_j(\mathbf{r};t) \right]^2 + \\
&+ \sum_{jpkq} \rho_{jpkq}(t) \left[\int d\mathbf{r} \varphi_j^*(\mathbf{r};t) \hat{a}(\mathbf{r}) \varphi_k(\mathbf{r};t) \right] \left[\int d\mathbf{r} \varphi_p^*(\mathbf{r};t) \hat{a}(\mathbf{r}) \varphi_q(\mathbf{r};t) \right] \Bigg\},
\end{aligned} \tag{12}$$

where $\{\varphi_j(\mathbf{r};t)\}$ and $\{\rho_j(t)\}$ are, respectively, the natural orbitals and occupation numbers, and $\rho_{jpkq}(t)$ the elements of the reduced two-body density matrix (2-RDM), $\rho^{(2)}(\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_1', \mathbf{r}_2'; t) = \sum_{jpkq} \rho_{jpkq}(t) \varphi_j^*(\mathbf{r}_1'; t) \varphi_p^*(\mathbf{r}_2'; t) \varphi_k(\mathbf{r}_1; t) \varphi_q(\mathbf{r}_2; t)$. For one-body operators which are local in position space Eq. (12) boils down to [15]

$$\frac{1}{N}\Delta^2_{\hat{A}}(t) = \int d\mathbf{r}\frac{\rho(\mathbf{r};t)}{N}\hat{a}^2(\mathbf{r}) - N\left[\int \frac{\rho(\mathbf{r};t)}{N}\hat{a}(\mathbf{r})\right]^2 + \\ + \int d\mathbf{r}_1 d\mathbf{r}_2 \frac{\rho^{(2)}(\mathbf{r}_1,\mathbf{r}_2,\mathbf{r}_1,\mathbf{r}_2;t)}{N} a(\mathbf{r}_1)a(\mathbf{r}_2), \tag{13}$$

where $\rho(\mathbf{r};t)$ is the density, i.e., the diagonal part of the 1-RDM Eq. (5). The density is one of the most basic quantities of a bosonic system and used among others to define the survival probability for the dynamics in double-well potentials, see below.

In the following, the variances of the position [$\hat{a}(\mathbf{r}) = \hat{x}$ or $\hat{a}(\mathbf{r}) = \hat{y}$], momentum [$\hat{a}(\mathbf{r}) = \hat{p}_x$ or $\hat{a}(\mathbf{r}) = \hat{p}_y$], and angular-momentum [$\hat{a}(\mathbf{r}) = \hat{x}\hat{p}_y - \hat{y}\hat{p}_x$] many-particle operators are used. Via these variances, we analyze and classify correlations in trapped finite BECs and at the limit of an infinite-number of particles [19, 53–58]. The latter case is in particular interesting since the bosons are known among other properties to be 100% condensed [59, 60, 94–100].

4 Photoionization of Neon

Atoms and molecules interact with light. Matter-light interaction is a central topic for technologies that affect our daily life, from energy production in solar panels to X-ray imaging in medicine or simply a pair of sunglasses to enjoy the summer. At a more fundamental level, matter-light interaction is at the origin of many biological processes from the process of vision to the life of photosynthetic organisms. These different examples have in common one physical process, namely, the absorption of an electromagnetic field – the incident light – by an atom or molecule. The atoms are made of a positively charged nucleus surrounded by a certain number of negatively charged electrons, bound together by the Coulomb interaction; molecules are a complex arrangement of atoms. The microscopic world follows the rules of quantum mechanics, dictated by the Schrödinger equation, which has a known analytical solution for only a handful of systems. Describing matter-light interaction at the level of a single atom or molecule is a challenging numerical task; methods such as MCTDH-X are developed to properly address this problem.

In this work [15], we discuss the specific case of photoionization, one of the outcomes of matter-light interaction. In this process, the atom absorbs the electromagnetic field, and the gained excess of energy pulls out an electron which is freed and escapes the atom. The probability that such an event happens is associated with the photoionization cross section. This quantity, among others, has the advantage to be accessible experimentally and numerically from methods in the MCTDH-X family [1–11, 28–32, 101–103], for instance. Thus, photoionization cross section is a testbed to assess the accuracy of numerical descriptions of matter-light interaction [68, 104]. An illustration of how accurate MCTDH-X-like methods can be in comparison with experimental results [105, 106] is provided in Fig. 3 for Neon atoms for different energies of the light.

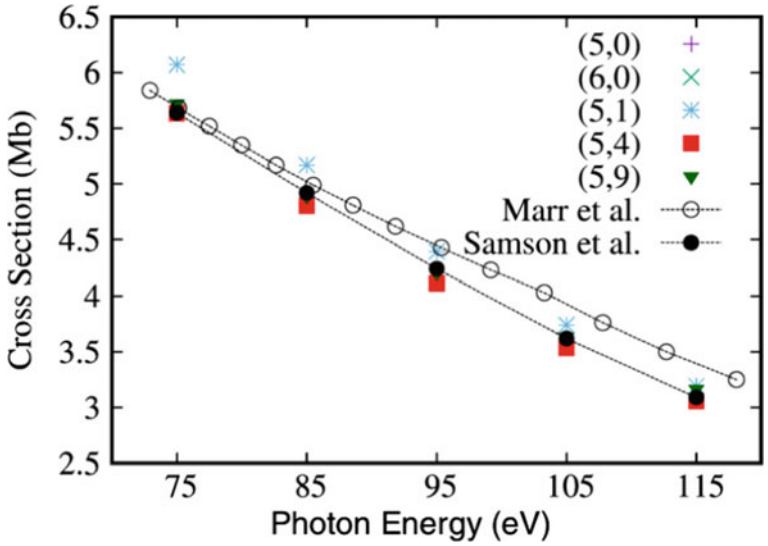


Fig. 3 Photoionization cross section (Eq. (11)) for neon computed using RAS-MCTDH-X for a ten-cycle pulse with $10^{14}\,\frac{W}{cm^2}$ peak intensity in comparison with experimental results [105, 106]. The (N_1, N_2) legend indicates the RAS scheme applied: N_1 and N_2 orbitals are used to build the $\mathcal{P}_1$ and $\mathcal{P}_2$ subspaces, respectively (see Fig. 2). Figure reprinted from [15] with data from [68].

5 Hierarchical Superfluidity in Bose-Einstein Condensates Interacting with Light

From our daily life, we are familiar with three different states of matter: solid, liquid, and gas. At very high temperatures, atoms and electrons form a soup of particles known as plasma, the fourth state of matter. At very low temperatures, instead, neutral atoms whose nuclei have an even number of neutrons (called 'bosons') can reach a fifth state of matter known as a Bose-Einstein condensate (BEC) [107–109]. In a BEC, all the atoms are in the same coherent quantum state and therefore behave as a whole entity. This coherence grants them *superfluid* properties: they can act as a fluid with zero viscosity and friction.

Things become even more interesting when we let the BEC interact with light. Experimentally, this can be achieved by placing the BEC inside an optical cavity (i.e., between two highly reflective mirrors) and shining a laser onto it. As the photons from the laser bounce back and forth between the two mirrors, they can resonantly couple to the atoms in the BEC and generate an effective optical lattice potential, an energetic landscape of peaks and valleys that the atoms have to adjust to [90, 110–112]. This can lead to a plethora of new phenomena, such as self-organization of the atoms in lattice structures that can even break the superfluid coherence and push the atoms into a different state called a self-organized Mott insulator (SMI) [110, 113–116].

In this work [16], we use MCTDH-X [14] to calculate density distributions and atomic correlations [84], and investigate the crossover between superfluidity and Mott insulation for a setup in which both the atoms and the cavity are blue-detuned (i.e., with their respective frequencies slightly lower than the resonant frequency) from the external pumping laser (Fig. 4a). Previous studies [117, 118] had already highlighted the emergence of dynamical instabilities in the blue-detuned regime. Our results unveil the mechanism that leads to such dynamical instabilities: a hierarchical self-organization of the atoms, corresponding to a transition from a single-well optical lattice to a double-well optical lattice (see Fig. 4c). This hierarchical self-organization induces two new kinds of phases with local superfluid correlations, but differing global properties (Fig. 4b). In the self-organized dimerized superfluid (SDSF) phase, coherence is still retained across the double-well dimers (Fig. 5e–g). In the second-order superfluid (2-SSF), instead, coherence persists only within each double-well dimer (Fig. 5h–j). Both of these higher-order correlated phases eventually erupt into chaotic behavior either directly (SDSF) or via quasiperiodic attractors (2-SSF) (Fig. 4b). Our results are able to explain the mechanism behind the same dynamical instabilities observed in the experiments [119], and illustrate the potential for blue detuning to realize exotic superfluid phases of matter.

6 Dynamical Behavior of Bosons and Fermions in a Double Well

In our daily life, cities and towns are sometimes separated apart from each other by hills and mountains. With a small hill, it is easy to build a pass or a tunnel to connect the communities from both sides, and the residents can communicate easily; while wide and tall mountains would prevent such communication. The same philosophy also applies for a quantum double-well system. In this system, two quantum wells are separated from each other by a barrier. Particles can hop from one well into another, and the probability of such hopping decreases with a wider and taller barrier. In a quantum system, the competition between the hopping strength and the particles' repulsive interactions determines the difficulty of the "communication" between distinct wells. When this communication is strong, coherence is established between the particles on distinct sides of the barrier – the system is in a *superfluid* phase; when there is no coherence, the system is in a *Mott insulator* phase [120–122]. The level of "communication" can be quantified by, for example, the Glauber one-body correlation function between the two wells [83, 84] (Eq. (6)). The transition between these two phases can be described by the Hubbard model [122].

In this work [17], we provide a step-by-step tutorial on the MCTDH-X software, and, as an example, we use it to investigate the dynamical behaviors of particles in a double-well system. We prepare the system without the central barrier and then ramp it up at different paces to a desired value. By extracting the correlation function [84], single-shot images [85, 90, 91], and occupation order parameter [85] of the system,

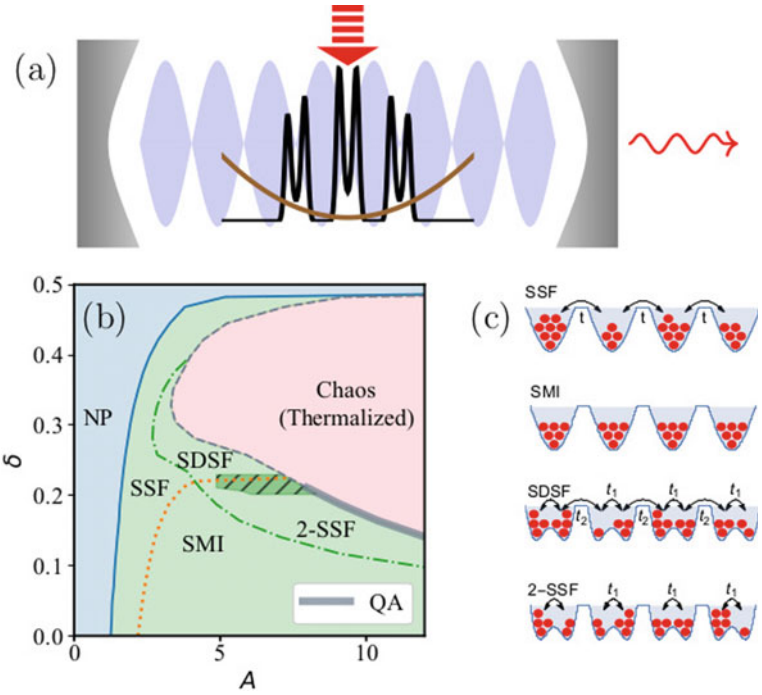


Fig. 4 (a) Schematic of a trapped cavity-BEC. (b) Phase diagram. For cavity detuning $0 < \delta < 1/2$, the system transitions from the normal phase to the dynamically unstable region via superradiance with increasing pump rate $A \propto \eta$. The strongly correlated phases are a self-organized superfluid (SSF), a self-organized Mott insulator (SMI), a self-organized dimerized superfluid (SDSF), and a self-organized second-order superfluid (2-SSF) phase. The orange line delineates superfluid phases and globally Mott insulating phases, whereas the green line marks the hierarchical self-organization to dimerized phases. Pronounced sensitivity to the ramping protocol is seen in the hatched dark green region. At higher A, the system is dynamically unstable to the formation of quasiperiodic attractors (QA), followed by chaos. The QAs only exist in the region represented by the thick gray line, whereas the thin dashed section represents a direct transition to the chaos. (c) Sketch of the SSF, SMI, SDSF, and 2-SSF phases. Figure adapted from [16]. (Color figure online)

we observe that the system undergoes different kinds of transitions depending on the ramping protocol, and the number of particles.

With a slow ramp and an even number of particles, the correlation gradually diminishes as the system slowly transitions from the superfluid to Mott insulator phase (Fig. 6(a,c,f)). However, with an odd number of particles, quantum effects kick in. One of the particles straddles across the two wells, having 50% probability to be in either well. As a result, a large residual correlation remains between the two wells even with a strong barrier, and the superfluid-Mott insulator transition becomes of first order with a clear hysteretical characteristics (Fig. 6(b,d,e)).

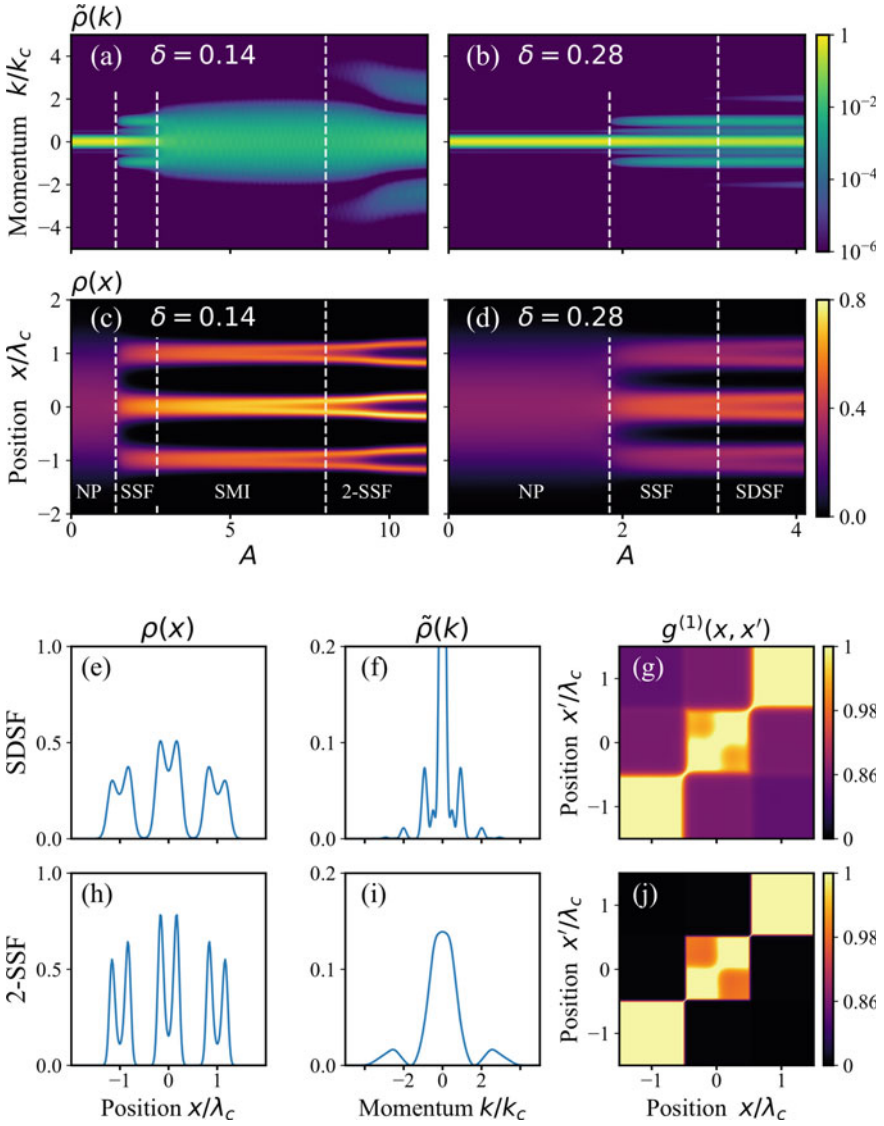


Fig. 5 (a)–(d) The momentum and position space density distributions $\tilde{\rho}(k)$ (Eq. (5) for $k' = k$) and $\rho(x)$ (Eq. (5) for $x' = x$) as a function of pump rate $A \propto \eta$ at two detunings $\delta = 0.14$ and $\delta = 0.28$. At lower detuning $\delta = 0.14$, the system starts from the normal phase and then enters the self-organized superfluid (SSF) phase, the SMI phase, and the 2-SSF phase sequentially. At higher detuning $\delta = 0.28$, the system starts from the normal phase, then enters the SSF phase, and the SDSF phase sequentially. The dotted lines are guides to the eye. (e)–(j) The position and momentum space density distributions and the Glauber one-body correlation function $g^{(1)}$ (Eq. (6)) of a SDSF state (first row) and a 2-SSF state (second row). In panels (g) and (j), the color code follows the function $-\ln(1 - g^{(1)})$. Note that the double-well splitting is seen in the central lattice site but not in the other two lattice sites because only $M = 4$ orbitals are used in the numerical simulations. Figure reprinted from [16].

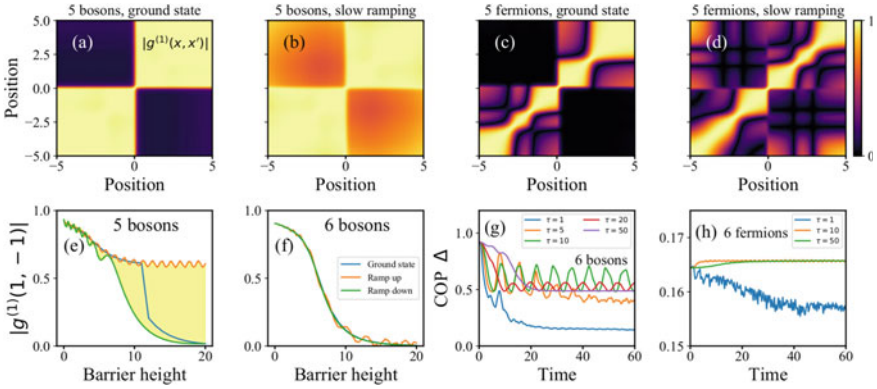


Fig. 6 (a–d) One-body correlation functions $|g^{(1)}(x, x')|$ (Eq. (6)) of $N = 5$ weakly-interacting bosons (a,b) and $N = 5$ weakly-interacting fermions (c,d). In (a,c) the ground states and in (b,d) slowly evolving states are shown. (e,f) The one-body correlations between the two wells $|g^{(1)}(1, -1)|$ as a function of barrier height (blue) in the ground state, when the barrier is ramped up (orange) and ramped down (green) slowly, for a system with (e) $N = 5$ and (f) $N = 6$ bosons. Hysteresis is clearly seen and marked in yellow in panel (e) but absent in panel (f). The ramping time is chosen as $\tau = 100$ for all cases in panels (b), (d), (e) and (f), and the correlation functions at $t = 100$ are shown in panels (b) and (d). (g,h) Occupation order parameter Δ (Eq. (7)) of (g) $N = 6$ bosons and (h) $N = 6$ fermions as a function of time with different ramping times τ. The dynamical behavior exhibits two different regimes, one for $\tau \lesssim 10$ and the other for $\tau \gtrsim 10$ for bosons and fermions alike, implying a dynamical phase transition at $\tau \simeq 10$. Figure reprinted from [17]. (Color figure online)

More interesting phenomena can be seen when the ramping rate differs, even with an even number of particles. In the slow ramping rate limit, the system becomes steady immediately after the ramping stops (Fig. 6(g,h) and 7(c,f,i,l)). As the ramp becomes faster, an oscillation of the system can be seen on top of this steady state, whose magnitude increases with the ramping rate (Fig. 6(g,h) and 7(b,e,h,k)). However, the system undergoes a dynamical transition as the ramping rate becomes fast enough, and instead of oscillating, the system becomes fluctuative (Fig. 6(g,h) and 7(a,d,g,j)). These transitions can be seen with particles of both statistics, i.e., bosons and fermions.

Our tutorial [17] thus investigates two kinds of dynamical phase transitions in a double well system. These physically intuitive examples provide a first-hand impression and understanding of the functionality of MCTDH-X.

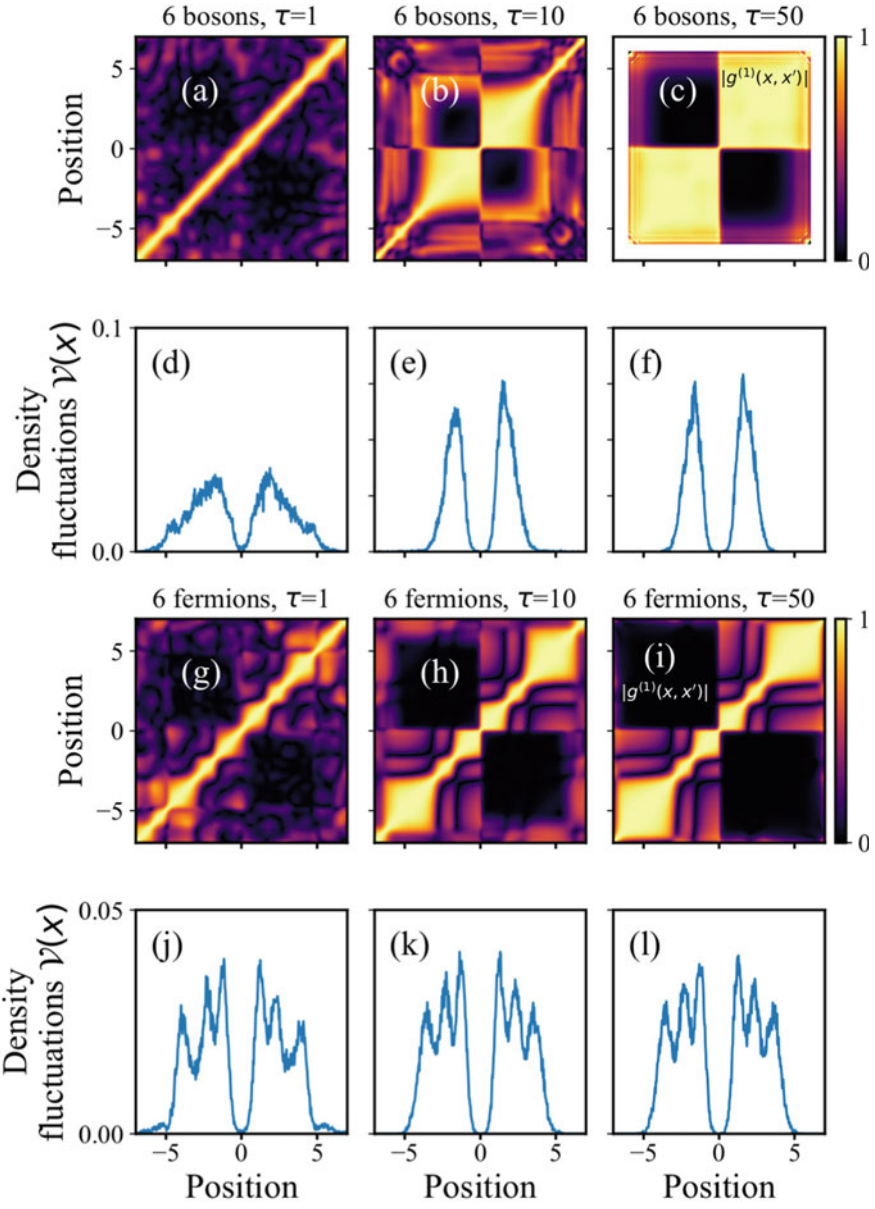


Fig. 7 One-body correlation functions $|g^{(1)}(x, x')|$ (Eq. (6)) (a–c) and variance over 10, 000 single-shot experiments (Eq. (10)) (d–f) of $N = 6$ bosons at $t = 80$ with different ramping times $\tau = 1, 10, 50$. At fast ramping, $\tau = 1$, the correlations $|g^{(1)}|$ are fluctuative. In the bosonic system, the distribution of the single-shot variance depends significantly on the ramping rate. The same quantities as in (a–f), but for fermions (g–l). The one-body correlation functions have quite a similar structure to those of the bosons [compare panels (g–i) and (a–c)]. Figure reprinted from [17].

7 Spectral Structure and Many-Body Dynamics of Ultracold Bosons in a Double-Well

During the last decades, the experimental manipulation of interacting quantum many-body systems has seen astounding advances, in particular in ultra-cold atoms setups, where a well-defined number of particles can be loaded in magnetic-optical traps [123, 124]. The high tunability of the confinement potential and inter-particle interactions has led to the investigation of out-of-equilibrium many-body phenomena, like higher-order tunneling and the build up of quantum correlations. Studying these features when excitations beyond the first energy band play a role becomes a massive numerical challenge even if one considers text-book examples like a double-well confining potential. In the specific case of a double well, the restriction to the first energy band in the physical description is termed two-mode approximation. While several extensions of the two-mode approximation have been investigated [126, 127], ultimately more powerful numerical schemes relying on a much less restrictive basis set must be employed to describe accurately the dynamics of the particles, such as the time-dependent density matrix renormalization group [125] or MCTDH-X methods [1–3, 15, 28–32].

In our contribution [18], we have analyzed the spectral and dynamical properties of bosons in a switchable symmetric double-well, for a tunable inter-particle interaction strength and different initial conditions. We employ the MCDTH-X method for the investigation of the Hamiltonian of $N \leq 10$ ultracold bosons with repulsive contact interactions. The bosons are loaded into a one-dimensional double-well potential with a time-dependent central barrier whose amplitude is ramped up linearly.

We explore how the well-understood single-particle spectrum is modified by adding a second and, subsequently, a third identical particle, based on an exact diagonalization of the many-particle Hamiltonian. In contrast to previous studies [130, 131], we do not focus only on the low-lying spectrum, but also on higher excitations, e.g., at the saddle-point of the confining potential.

When considering a time-independent potential, we showcase the breakdown of the two-mode approximation for sufficiently large interaction strengths when two particles are launched in one of the wells at the single-particle ground state. As a result, we are able to characterize the nature of the correlated tunneling process for weak interactions: While on first sight the detection probabilities in each well seem to indicate a direct, first-order pairwise tunneling process, we find that the tunneling is actually described by a second-order process. This observation is possible thanks to the numerical methods that allow for the examination of the time-integrated probability current.

After the insightful study of the spectral properties of the system, we proceeded to tackle the time-dependent scenario with MCDTH-X. We focus on the dynamics of the von Neumann entropy (8) for quantifying the non-separability of the many-particle state and aim to understand how the excitations spread over the many-particle basis. The MCDTH-X method lets us calculate the time evolution of the von Neumann entropy when varying the ramping time of the potential barrier T_{ramp}, for small and

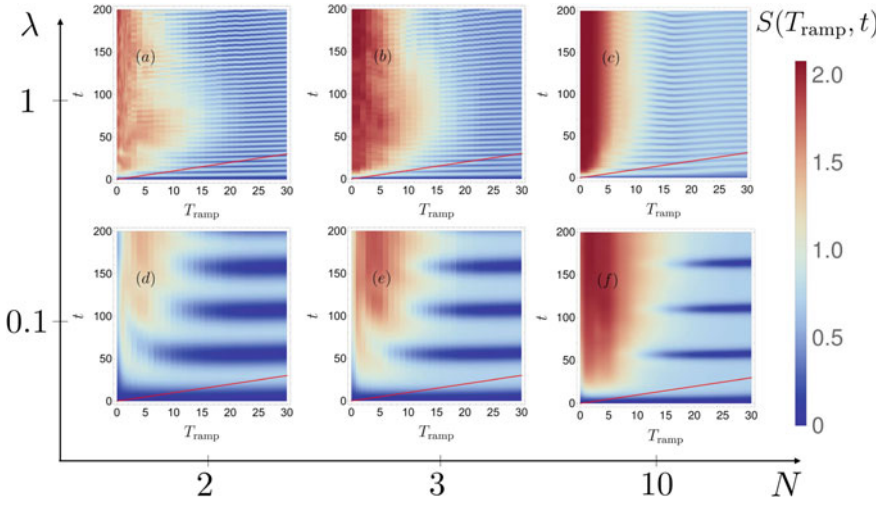


Fig. 8 Time evolution of the von Neumann entropy $S(T_{\mathrm{ramp}}, t)$, Eq. (8), as a function of the ramping time T_{ramp} for increasing particle number from left to right, and two different values of the interaction strength λ (rows). The red line in each panel indicates the full switching duration $t = T_{\mathrm{ramp}}$ for the ramp to reach its maximum. Large values of S indicate that many many-particle states are populated. Coherent oscillations between the ground state and the first excited state are observed for $T_{\mathrm{ramp}} > 20$. Figure reprinted from Ref. [18]. (Color figure online)

large interaction ($\lambda = 0.1$ and 1), and for an increasing number of bosons ($N = 2$, 3, and 10), see Fig. 8. We have found a cross-over from diabatic to quasi-adiabatic evolution when increasing the ramping time T_{ramp}.

Based on our aforementioned characterization of the spectrum, we find that diabatic switching leads to an efficient energy transfer through the population of a large number of many-particle excited states while a (quasi-) adiabatic ramp only populates lowest-lying excited states. As evident from Fig. 8, those coarse-grained features emerging in the time evolution of the entropy are robust as the particle number is increased from two to ten.

Our study exemplifies the potential and flexibility of the MCDTH-X method for investigating the many-body dynamics of interacting bosons subject to a variety of numerically challenging scenarios such as finite-time switching of the double well's potential barrier.

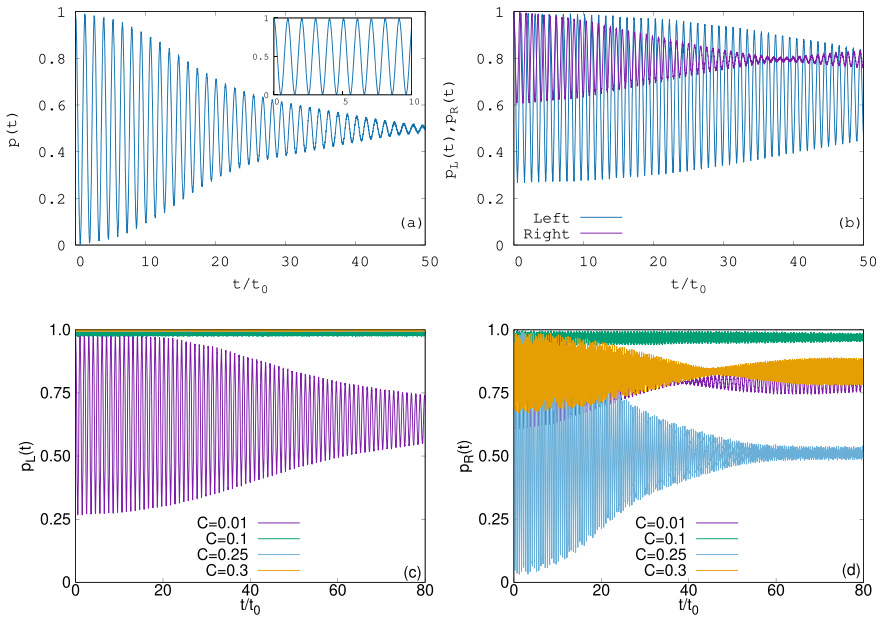


Fig. 9 (a) Many-body time evolution of the survival probability $p(t)$ in a symmetric double well computed with $M = 2$ time-adaptive orbitals. The number of bosons is $N = 1000$ and the interaction parameter $\Lambda = \lambda_0(N-1) = 0.1$. The respective mean-field dynamics is shown in the inset. (b) Many-body results for the left $[p_L(t)]$ and right $[p_R(t)]$ survival probabilities in an asymmetric double well with a small asymmetry $C = 0.01$ and in panels (c) and (d) for different asymmetries. Fragmented resonant tunneling is found for $p_R(t)$ and asymmetry $C = 0.25$ [51]. A common time unit t_0 is chosen for plotting the results for double wells with different asymmetries C. The quantities shown are dimensionless. Figure panels reprinted from [51].

8 Universality of Fragmentation and Fragmented Resonant Tunneling in an Asymmetric Bosonic Josephson Junction

The out-of-equilibrium quantum dynamics of interacting bosons trapped in a one-dimensional asymmetric double-well potential is studied by solving the many-body Schrödinger equation numerically accurately using the MCTDHB method [2, 4, 15]. Our aim is to examine how the gradual loss of the reflection symmetry of the confining trap potential affects the macroscopic quantum tunneling dynamics of the interacting bosons between the two wells. In our asymmetric double well, the left well is deeper than the right one. This implies that the left, $p_L(t) = \int_{-\infty}^{0} dx \frac{\rho(x;t)}{N}$, and right, $p_R(t) = \int_{0}^{+\infty} dx \frac{\rho(x;t)}{N}$, survival probabilities, computed by initially preparing the condensate in the left and right wells, respectively, are generally different. This is unlike the case of the symmetric double well for which obviously $p_L(t) = p_R(t) \equiv p(t)$.

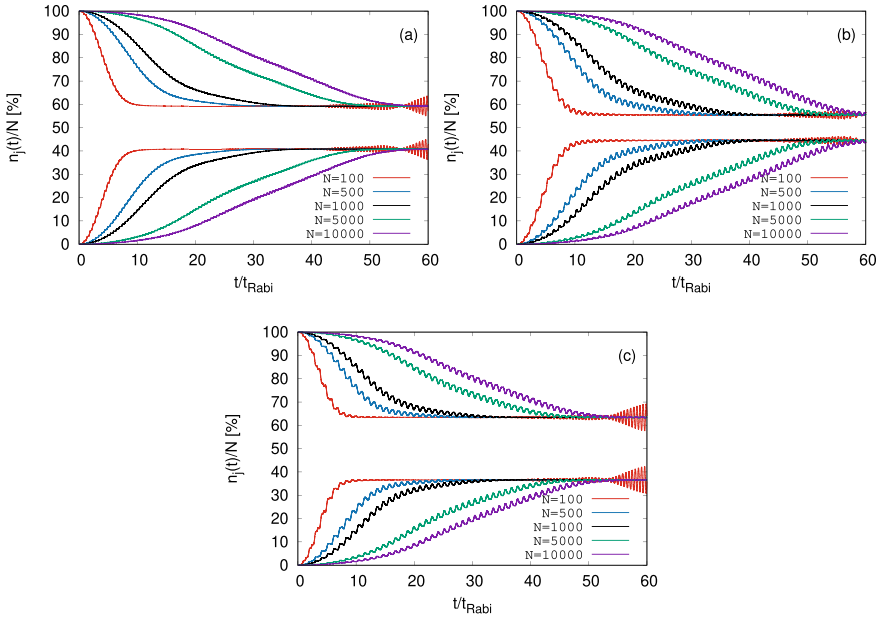


Fig. 10 Universality of fragmentation in an asymmetric bosonic Josephson junction. The interaction parameter is $\Lambda = \lambda_0(N-1) = 0.1$ and the number of time-adaptive orbitals in $M = 2$. (a) The natural occupation numbers $\frac{n_1(t)}{N}$ (upper curves) and $\frac{n_2(t)}{N}$ (lower curves) for BECs consisting of a different number of bosons N in a symmetric double well. (b) [(c)] Same as panel (a) but for an asymmetric double well with asymmetry $C = 0.001$ and for preparing the bosons in the left [right] well. The universal fragmentation value increases (decreases) when the bosons are initially prepared in the left (right) well for $C = 0.001$ [51]. The quantities shown are dimensionless. Figure panels reprinted from [51].

The dynamics of the bosons is studied by analyzing physical quantities of increasing many-body complexity, i.e., the survival probability, depletion and fragmentation, and the many-particle position and momentum variances. Explicitly, we have examined the oscillatory behavior and decay of the survival probabilities, the pace of development and degree of fragmentation, and the growth and differences to mean field of the many-particle position and momentum variances. We find an overall suppression of the oscillations of the survival probabilities in an asymmetric double well, except for resonant values of the asymmetry, see Fig. 9d for asymmetry $C = 0.25$ for which the one-body ground state energy in the right well matches the one-body first excited state energy in the left well.

As a general rule, we observe that $p_L(t)$ and $p_R(t)$ are affected differently by the repulsive boson-boson interaction. For a sufficiently strong repulsive interaction, the BEC fragments while oscillating in the asymmetric junction. The degree of fragmentation depends in an intricate manner both on the asymmetry C and the initial well in which the BEC is prepared. Chiefly, we discover that resonant tunneling of interacting bosons in accompanied by depletion and eventually fragmentation. Last

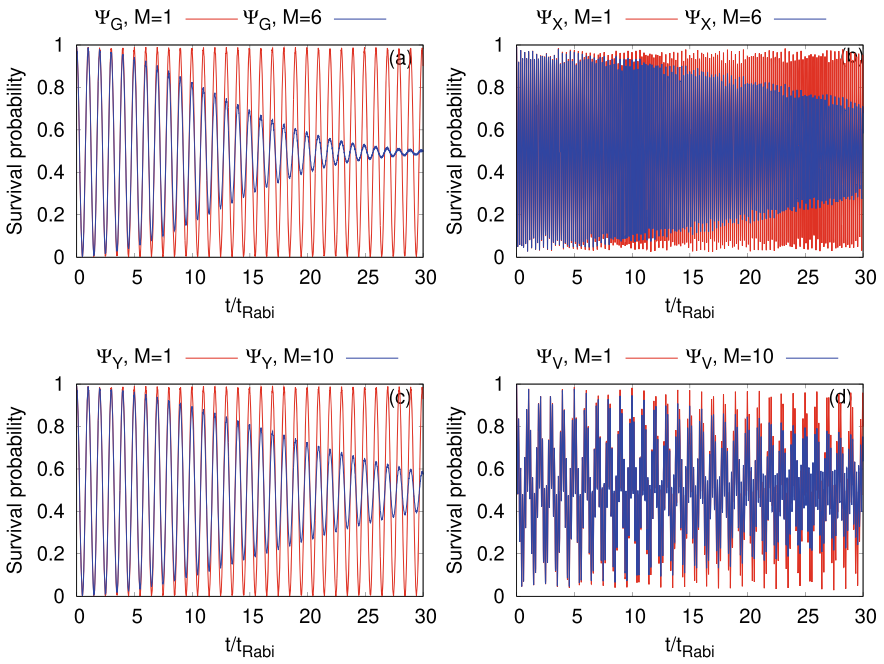


Fig. 11 Time evolution of the survival probability of $N = 10$ bosons in the left well of a symmetric two-dimensional double-well potential for the initial states (a) $\Psi_G(\mathbf{r})$, (b) $\Psi_X(\mathbf{r})$, (c) $\Psi_Y(\mathbf{r})$, and (d) $\Psi_V(\mathbf{r})$. The interaction parameter is $\Lambda = \lambda_0(N-1) = 0.01\pi$. Many-body results (curves in blue) and corresponding mean-field results (curves in red). We used $M = 6$ time-adaptive orbitals for $\Psi_G(\mathbf{r})$ and $\Psi_X(\mathbf{r})$ and $M = 10$ time-adaptive orbitals for $\Psi_Y(\mathbf{r})$ and $\Psi_V(\mathbf{r})$. Objects which include transverse excitations lose coherence faster although their density oscillations decay slower, see [20] for more details. The quantities shown are dimensionless. Figure panels reprinted from [20]. (Color figure online)

but not least, we establish that the universality of fragmentation previously found in symmetric double wells [61] is robust to the trap asymmetries, see Fig. 10b,c. The universality of fragmentation implies a macroscopically-large fragmented BEC in the junction whose many-particle position variance per particle, $\frac{1}{N}\Delta^2_{\hat{X}}(t)$, diverges at the limit of an infinite number of particles [51].

9 Impact of the Transverse Direction on the Many-Body Tunneling Dynamics in a Two-Dimensional Bosonic Josephson Junction

Tunneling in a many-body system is one of the novel implications of quantum mechanics where particles move in a region of space under a classically-forbidden

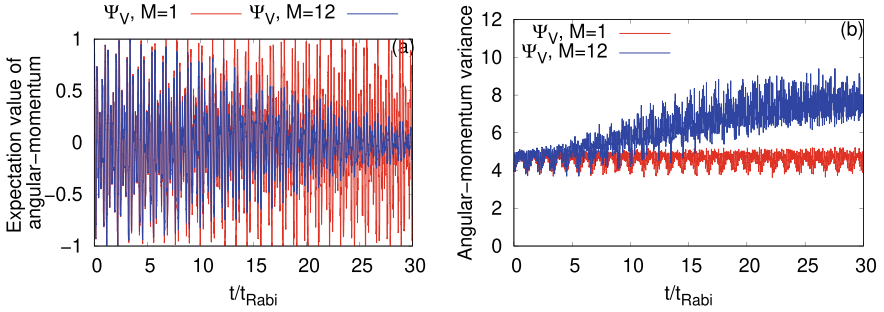


Fig. 12 (a) Expectation value per particle of the angular momentum operator $\frac{1}{N}\langle\Psi_V|\hat{L}_Z|\Psi_V\rangle$, and (b) variance per particle of the angular momentum operator, $\frac{1}{N}\Delta^2_{\hat{L}_Z}(t)$, for the vortex state $\Psi_V(\mathbf{r})$ in a symmetric two-dimensional double-well potential. The number of bosons is $N = 10$ and the interaction parameter $\Lambda = \lambda_0(N-1) = 0.01\pi$. Many-body ($M = 12$ time-adaptive orbitals; curves in blue) and mean-field ($M = 1$ time-adaptive orbitals; curves in red) results. The decay of the expectation value of the angular-momentum operator and the growth of its variance are seen to accompany the fragmentation of the vortex state, see [20] for more details. The quantities shown are dimensionless. Figure panels reprinted from [20]. (Color figure online)

potential barrier. In our investigation we theoretically describe the out-of-equilibrium many-body tunneling dynamics of a few structured bosonic clouds in a two-dimensional symmetric double-well potential using the MCTDHB method [2, 4, 15]. The motivation for the research is to unravel how the inclusion of the transverse y direction, orthogonal to the junction of the double-well along x direction, influences the tunneling dynamics of the bosonic clouds.

Explicitly, in [20] we investigate the tunneling dynamics by preparing the initial state of the bosonic clouds in the left well of the double-well either as the ground $[\Psi_G(\mathbf{r})]$, longitudinally $[\Psi_X(\mathbf{r})]$ or transversely $[\Psi_Y(\mathbf{r})]$ excited, or as a vortex state $[\Psi_V(\mathbf{r})]$. We elucidate the detailed mechanism of the tunneling process by analyzing the evolution in time of the survival probability, depletion and fragmentation, and the expectation values and variances of the many-particle position, momentum, and angular-momentum operators. We find, as a general rule, that all objects lose coherence while tunneling under the potential barrier and that the states which include transverse excitations, $\Psi_Y(\mathbf{r})$ and $\Psi_V(\mathbf{r})$, do so quicker than the states that do not include transverse excitations, $\Psi_G(\mathbf{r})$ and $\Psi_X(\mathbf{r})$, respectively. This is despite the slower decay rate of the density oscillations of the former with respect to the latter, see Fig. 11 and [20] for more details.

Unlike our previous work on tunneling of a vortex state in a circular double-well trap [62], for which the angular momentum is naturally a good quantum number, in the standard two-dimensional double-well trap angular momentum is not a good quantum number. For a mean-field dynamics of a vortex state in the later trap see [132]. Our study puts forward preliminary results for exploring and identifying the many-body rules of transport of angular momentum and its variance in trapped matter-wave systems, see Fig. 12.

10 Summary and Outlook

The wealth and reach of the contributions reporting results from the application of the MCTDH-X software implementations [12–14] on the high-performance computation facilities at the HLRS in Stuttgart is increasing. We found and described the exciting fundamental many-body physics of novel hierarchically-superfluid quantum phases of bosons immersed in high-finesse optical cavities [16], of correlated bosons' and fermions' dynamics in double-well potentials [17], the dynamics and spectral features of entropy resulting from quenches for bosons in double-well potentials [18], the universality of fragmentation in an asymmetric bosonic Josephson junction [19], as well as the effect of the transverse direction on the tunneling dynamics in a two-dimensional bosonic Josephson junction [20]. MCTDH-X, RAS-MCTDH-X and the expanding field of their applications were reviewed in the *Reviews of Modern Physics* [15]. The activities of our MCTDH-X project has delivered an enormous boost of the toolbox to analyze the quantum many-body physics of indistinguishable particles. Our novel tools enable, for instance, investigations of the position and momentum space variances, the density fluctuations, and the phase diagrams of ultracold atoms with an unprecedented detail. We anticipate that another series of significant and important findings will result from the application of our new set of many-body analysis tools in the forthcoming years.

Acknowledgments Financial support by the Deutsche Forschungsgemeinschaft (DFG) is gratefully acknowledged. OEA acknowledges funding by the Israel Science Foundation (Grants No. 600/15 and No. 1516/19). We acknowledge financial support by the Austrian Science Foundation (FWF) under grants P32033 and M2653. LdFdP acknowledges the Alexander von Humboldt-Foundation for financial support. MABM acknowledges financial support from CONACyT postdoctoral fellowship program. We acknowledge financial support by the Swiss National Science Foundation (SNSF), the NCCR Quantum Science and Technology, and Mr. G. Anderheggen.

References

1. A.I. Streltsov, O.E. Alon, L.S. Cederbaum, Phys. Rev. A **73**, 063626 (2006)
2. A.I. Streltsov, O.E. Alon, L.S. Cederbaum, Phys. Rev. Lett. **99**, 030402 (2007)
3. O.E. Alon, A.I. Streltsov, L.S. Cederbaum, Chem. Phys. **127**, 154103 (2007)
4. O.E. Alon, A.I. Streltsov, L.S. Cederbaum, Phys. Rev. A **77**, 033613 (2008)
5. K. Sakmann, A.I. Streltsov, O.E. Alon, L.S. Cederbaum, Phys. Rev. Lett. **103**, 220601 (2009)
6. A.U.J. Lode, K. Sakmann, O.E. Alon, L.S. Cederbaum, A.I. Streltsov, Phys. Rev. A **86**, 063606 (2012)
7. A.U.J. Lode, Phys. Rev. A **93**, 063601 (2016)
8. *Multidimensional Quantum Dynamics: MCTDH Theory and Applications*, ed. by H.-D. Meyer, F. Gatti, G.A. Worth (Wiley-VCH, Weinheim, 2009)
9. K. Sakmann, *Many-Body Schrödinger Dynamics of Bose-Einstein Condensates, Springer Theses* (Springer, Heidelberg, 2011)
10. *Quantum Gases: Finite Temperature and Non-equilibrium Dynamics*, ed. by N.P. Proukakis, S.A. Gardiner, M.J. Davis, M.H. Szymanska, Cold Atoms Series, vol. 1 (Imperial College Press, London, 2013)

11. A.U.J. Lode, *Tunneling Dynamics in Open Ultracold Bosonic Systems, Springer Theses* (Springer, Heidelberg, 2015)
12. A.I. Streltsov, L.S. Cederbaum, O.E. Alon, K. Sakmann, A.U.J. Lode, J. Grond, O.I. Streltsova, S. Klaiman, R. Beinke, *The Multiconfigurational Time-Dependent Hartree for Bosons Package*, version 3.x. http://mctdhb.org, Heidelberg/Kassel (2006-Present)
13. A.I. Streltsov, O.I. Streltsova, *The multiconfigurational time-dependent Hartree for bosons laboratory*, version 1.5. http://MCTDHB-lab.com
14. A.U.J. Lode, M.C. Tsatsos, S.E. Weiner, E. Fasshauer, R. Lin, L. Papariello, P. Molignini, C. Lévêque, MCTDH-X: the time-dependent multiconfigurational Hartree for indistinguishable particles software (2020). http://ultracold.org
15. A.U.J. Lode, C. Lévêque, L.B. Madsen, A.I. Streltsov, O.E. Alon, Rev. Mod. Phys. **92**, 011001 (2020)
16. R. Lin, P. Molignini, A.U.J. Lode, R. Chitra, Phys. Rev. A **101**, 061602(R) (2020)
17. R. Lin, P. Molignini, L. Papariello, M.C. Tsatsos, C. Lévêque, S.E. Weiner, E. Fasshauer, R. Chitra, A.U.J. Lode, Quantum Sci. Technol. **5**, 024004 (2020)
18. F. Schäfer, M.A. Bastarrachea-Magnani, A.U.J. Lode, L. de Forges de Parny, A. Buchleitner, Entropy **22**, 382 (2020)
19. O.E. Alon, Symmetry **11**, 1344 (2019)
20. A. Bhowmik, S.K. Halder, O.E. Alon, Sci. Rep. **10**, 21476 (2020)
21. A.U.J. Lode, K. Sakmann, R.A. Doganov, J. Grond, O.E. Alon, A.I. Streltsov, L.S. Cederbaum, in *High Performance Computing in Science and Engineering '13: Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2013*, ed. by W.E. Nagel, D.H. Kröner, M.M. Resch (Springer, Heidelberg, 2013), pp. 81–92
22. S. Klaiman, A.U.J. Lode, K. Sakmann, O.I. Streltsova, O.E. Alon, L.S. Cederbaum, A.I. Streltsov, in *High Performance Computing in Science and Engineering '14: Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2014*, ed. by W.E. Nagel, D.H. Kröner, M.M. Resch (Springer, Heidelberg, 2015), pp. 63–86
23. O.E. Alon, V.S. Bagnato, R. Beinke, I. Brouzos, T. Calarco, T. Caneva, L.S. Cederbaum, M.A. Kasevich, S. Klaiman, A.U.J. Lode, S. Montangero, A. Negretti, R.S. Said, K. Sakmann, O.I. Streltsova, M. Theisen, M.C. Tsatsos, S.E. Weiner, T. Wells, A.I. Streltsov, in *High Performance Computing in Science and Engineering '15: Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2015*, ed. by W.E. Nagel, D.H. Kröner, M.M. Resch (Springer, Heidelberg, 2016), pp. 23–50
24. O.E. Alon, R. Beinke, L.S. Cederbaum, M.J. Edmonds, E. Fasshauer, M.A. Kasevich, S. Klaiman, A.U.J. Lode, N.G. Parker, K. Sakmann, M.C. Tsatsos, A.I. Streltsov, in *High Performance Computing in Science and Engineering '16: Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2016*, ed. by W.E. Nagel, D.H. Kröner, M.M. Resch (Springer, Heidelberg, 2016), pp. 79–96
25. O.E. Alon, R. Beinke, C. Bruder, L.S. Cederbaum, S. Klaiman, A.U.J. Lode, K. Sakmann, M. Theisen, M.C. Tsatsos, S.E. Weiner, A.I. Streltsov, in *High Performance Computing in Science and Engineering '17: Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2017*, ed. by W.E. Nagel, D.H. Kröner, M.M. Resch (Springer, Heidelberg, 2018), pp. 93–115
26. O.E. Alon, V.S. Bagnato, R. Beinke, S. Basu, L.S. Cederbaum, B. Chakrabarti, B. Chatterjee, R. Chitra, F.S. Diorico, S. Dutta, L. Exl, A. Gammal, S.K. Haldar, S. Klaiman, C. Lévêque, R. Lin, N.J. Mauser, P. Molignini, L. Papariello, R. Roy, K. Sakmann, A.I. Streltsov, G.D. Telles, M.C. Tsatsos, R. Wu, A.U.J. Lode, in *High Performance Computing in Science and Engineering '18: Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2018*, ed. by W.E. Nagel, D.H. Kröner, M.M. Resch (Springer, Heidelberg, 2019), pp. 89–110
27. A.U.J. Lode, O.E. Alon, L.S. Cederbaum, B. Chakrabarti, B. Chatterjee, R. Chitra, A. Gammal, S.K. Haldar, M.L. Lekala, C.Lévêque, R. Lin, P. Molignini, L. Papariello, M.C. Tsatsos, *High Performance Computing in Science and Engineering '19: Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2021*, pp. 77–87
28. C. Lévêque, L.B. Madsen, New J. Phys. **19**, 043007 (2017)

29. H. Miyagi, L.B. Madsen, Phys. Rev. A **87**, 062511 (2013)
30. H. Miyagi, L.B. Madsen, J. Chem. Phys. **140**, 164309 (2014)
31. H. Miyagi, L.B. Madsen, Phys. Rev. A **89**, 063416 (2014)
32. H. Miyagi, L.B. Madsen, Phys. Rev. A **95**, 023415 (2017)
33. J. Olsen, B.O. Roos, P. Jørgensen, H.J.A. Jensen, J. Chem. Phys. **89**, 2185 (1988)
34. P. Kramer, M. Saraceno, *Geometry of the Time-Dependent Variational Principle in Quantum Mechanics*, Lecture Notes in Physics, vol. 140. (Springer, Berlin, 1981)
35. P.A.M. Dirac, Math. Proc. Cambridge Philos. Soc. **26**, 376 (1930)
36. N.F. Mott, J. Frenkel, *Wave Mechanics: Advanced General Theory* (Clarendon Press, Oxford, 1934)
37. C. Lévêque, L.B. Madsen, J. Phys. B **51**, 155302 (2018)
38. J. Grond, A.I. Streltsov, A.U.J. Lode, K. Sakmann, L.S. Cederbaum, O.E. Alon, Phys. Rev. A **88**, 023606 (2013)
39. J. Grond, A.I. Streltsov, L.S. Cederbaum, O.E. Alon, Phys. Rev. A **86**, 063607 (2012)
40. O.E. Alon, A.I. Streltsov, K. Sakmann, A.U.J. Lode, J. Grond, L.S. Cederbaum, Chem. Phys. **401**, 2 (2012)
41. S. Klaiman, A.U.J. Lode, A.I. Streltsov, L.S. Cederbaum, O.E. Alon, Phys. Rev. A **90**, 043620 (2014)
42. A.U.J. Lode, M.C. Tsatsos, J.L. Temp, Phys. **181**, 171 (2015)
43. T. Wells, A.U.J. Lode, V.S. Bagnato, M.C. Tsatsos, J.L. Temp, Phys. **180**, 133 (2015)
44. U.R. Fischer, A.U.J. Lode, B. Chatterjee, Phys. Rev. A **91**, 063621 (2015)
45. A.U.J. Lode, B. Chakrabarti, V.K.B. Kota, Phys. Rev. A **92**, 033622 (2015)
46. S.E. Weiner, M.C. Tsatsos, L.S. Cederbaum, A.U.J. Lode, Sci. Rep. **7**, 40122 (2017)
47. A.U.J. Lode, S. Klaiman, O.E. Alon, A.I. Streltsov, L.S. Cederbaum, Phys. Rev. A **89**, 053620 (2014)
48. I. Březinová, J. Burgdörfer, A.U.J. Lode, A.I. Streltsov, L.S. Cederbaum, O.E. Alon, L.A. Collins, B.I. Schneider, J. Phys. Conf. Ser. **488**, 012032 (2014)
49. A.U.J. Lode, A.I. Streltsov, K. Sakmann, O.E. Alon, L.S. Cederbaum, Proc. Natl. Acad. Sci. **109**, 13521 (2012)
50. O.E. Alon, J. Phys. Conf. Ser. **1206**, 012009 (2019)
51. S.K. Haldar, O.E. Alon, New J. Phys. **21**, 103037 (2019)
52. S.K. Haldar, O.E. Alon, J. Phys. Conf. Ser. **1206**, 012010 (2019)
53. O.E. Alon, L.S. Cederbaum, Chem. Phys. **515**, 287 (2018)
54. O.E. Alon, Mol. Phys. **117**, 2108 (2019)
55. S. Klaiman, O.E. Alon, Phys. Rev. A **91**, 063613 (2015)
56. S. Klaiman, A.I. Streltsov, O.E. Alon, Phys. Rev. A **93**, 023605 (2016)
57. K. Sakmann, J. Schmiedmayer, arXiv:1802.03746v2 [cond-mat.quant-gas]
58. S. Klaiman, R. Beinke, L.S. Cederbaum, A.I. Streltsov, O.E. Alon, Chem. Phys. **509**, 45 (2018)
59. S. Klaiman, L.S. Cederbaum, Phys. Rev. A **94**, 063648 (2016)
60. L.S. Cederbaum, Phys. Rev. A **96**, 013615 (2017)
61. K. Sakmann, A.I. Streltsov, O.E. Alon, L.S. Cederbaum, Phys. Rev. A **89**, 023602 (2014)
62. R. Beinke, S. Klaiman, L.S. Cederbaum, A.I. Streltsov, O.E. Alon, Phys. Rev. A **92**, 043627 (2015)
63. C. Lévêque, L.B. Madsen, J. Chem. Phys. **150**, 194105 (2019)
64. J.J. Omiste, L.B. Madsen, J. Chem. Phys. **150**, 084305 (2019)
65. B. Chatterjee, M.C. Tsatsos, A.U.J. Lode, New J. Phys. **21**, 033030 (2019)
66. S. Bera, B. Chakrabarti, A. Gammal, M.C. Tsatsos, M.L. Lekala, B. Chatterjee, C. Lévêque, A.U.J. Lode, Sci. Rep. **9**, 17873 (2019)
67. J.H.V. Nguyen, M.C. Tsatsos, D. Luo, A.U.J. Lode, G.D. Telles, V.S. Bagnato, R.G. Hulet, Phys. Rev. X **9**, 011052 (2019)
68. J.J. Omiste, L.B. Madsen, Phys. Rev. A **97**, 013422 (2018)
69. O. Penrose, L. Onsager, Phys. Rev. **104**, 576 (1956)
70. O.E. Alon, L.S. Cederbaum, Phys. Rev. Lett. **95**, 140402 (2005)
71. R.W. Spekkens, J.E. Sipe, Phys. Rev. A **59**, 3868 (1999)

72. A.I. Streltsov, L.S. Cederbaum, N. Moiseyev, Phys. Rev. A **70**, 053607 (2004)
73. S. Zöllner, H.-D. Meyer, P. Schmelcher, Phys. Rev. A **74**, 063611 (2006)
74. P. Noziéres, D. Saint James, J. Phys. (Paris) **43**, 1133 (1982)
75. P. Noziéres, in *Bose-Einstein Condensation*, ed. by A. Griffin, D.W. Snoke, S. Stringari (Cambridge University Press, Cambridge, 1996), p. 15
76. E.J. Mueller, T.-L. Ho, M. Ueda, G. Baym, Phys. Rev. A **74**, 033612 (2006)
77. U.R. Fischer, P. Bader, Phys. Rev. A **82**, 013607 (2010)
78. Q. Zhou, X. Cui, Phys. Rev. Lett. **110**, 140407 (2013)
79. Y. Kawaguchi, Phys. Rev. A **89**, 033627 (2014)
80. S.-W. Song, Y.-C. Zhang, H. Zhao, X. Wang, W.-M. Liu, Phys. Rev. A **89**, 063613 (2014)
81. H.H. Jen, S.-K. Yip, Phys. Rev. A **91**, 063603 (2015)
82. A.R. Kolovsky, Phys. Rev. A **95**, 033622 (2017)
83. K. Sakmann, A.I. Streltsov, O.E. Alon, L.S. Cederbaum, Phys. Rev. A **78**, 023615 (2008)
84. R.J. Glauber, Phys. Rev. **130**, 2529 (1963)
85. B. Chatterjee, A.U.J. Lode, Phys. Rev. A **98**, 053624 (2018)
86. R. Roy, A. Gammal, M.C. Tsatsos, B. Chatterjee, B. Chakrabarti, A.U.J. Lode, Phys. Rev. A **97**, 043625 (2018)
87. D. Jaksch, C. Bruder, J.I. Cirac, C.W. Gardiner, P. Zoller, Phys. Rev. Lett. **81**, 3108 (1998)
88. I. Březinová, A.U.J. Lode, A.I. Streltsov, O.E. Alon, L.S. Cederbaum, J. Burgdörfer, Phys. Rev. A **86**, 013630 (2012)
89. R. Roy, C. Lévêque, A.U.J. Lode, A. Gammal, B. Chakrabarti, Quantum Rep. **1**, 304 (2019)
90. A.U.J. Lode, C. Bruder, Phys. Rev. Lett. **118**, 013603 (2017)
91. K. Sakmann, M. Kasevich, Nature Phys. **12**, 451 (2016)
92. E. Foumouo, G.L. Kamta, G. Edah, B. Piraux, Phys. Rev. A **74**, 063409 (2006)
93. L.B. Madsen, L.A. Nikolopoulos, P. Lambropoulos, Eur. Phys. J. D **10**, 67 (2000)
94. Y. Castin, R. Dum, Phys. Rev. A **57**, 3008 (1998)
95. E.H. Lieb, R. Seiringer, J. Yngvason, Phys. Rev. A **61**, 043602 (2000)
96. E.H. Lieb, R. Seiringer, Phys. Rev. Lett. **88**, 170409 (2002)
97. L. Erdős, B. Schlein, H.-T. Yau, Phys. Rev. Lett. **98**, 040404 (2007)
98. L. Erdős, B. Schlein, H.-T. Yau, Invent. Math. **167**, 515 (2007)
99. A. Michelangeli, A. Olgiati, Anal. Math. Phys. **7**, 377 (2017)
100. O.E. Alon, J. Phys. A **50**, 295002 (2017)
101. L. Cao, V. Bolsinger, S.I. Mistakidis, G.M. Koutentakis, S. Krönke, J.M. Schurer, P. Schmelcher, J. Chem. Phys. **147**, 044106 (2017)
102. L. Cao, S. Krönke, O. Vendrell, P. Schmelcher, J. Chem. Phys. **139**, 134103 (2013)
103. J. Caillat, J. Zanghellini, M. Kitzler, O. Koch, W. Kreuzer, A. Scrinzi, Phys. Rev. A **71**, 012712 (2005)
104. D.J. Haxton, K.V. Lawler, C.W. McCurdy, Phys. Rev. A **86**, 013406 (2012)
105. G.V. Marr, J.B. West, At. Data Nucl. Data Tables **18**, 497 (1976)
106. J.A. Samson, W.C. Stolte, J. Electron Spectrosc. Relat. Phenom. **123**, 265 (2002)
107. C.J. Pethick, H. Smith, *Bose-Einstein Condensation in Dilute Gases*, 2nd edn. (Cambridge University Press, Cambridge, 2008)
108. M.H. Anderson, J.R. Ensher, M.R. Matthews, C.E. Wieman, E.A. Cornell, Science **269**, 198 (1995)
109. K.B. Davis, M.-O. Mewes, M.R. Andrews, N.J. van Druten, D.S. Durfee, D.M. Kurn, W. Ketterle, Phys. Rev. Lett. **75**, 3969 (1995)
110. R. Lin, L. Papariello, P. Molignini, R. Chitra, A.U.J. Lode, Phys. Rev. A **100**, 013611 (2019)
111. A.U.J. Lode, F.S. Diorico, R. Wu, P. Molignini, L. Papariello, R. Lin, C. Lévêque, L. Exl, M.C. Tsatsos, R. Chitra, N.J. Mauser, New J. Phys. **20**, 055006 (2018)
112. P. Molignini, L. Papariello, A.U.J. Lode, R. Chitra, Phys. Rev. A **98**, 053620 (2018)
113. P. Domokos, H. Ritsch, Phys. Rev. Lett. **89**, 253003 (2002)
114. D. Nagy, G. Szirmai, P. Domokos, Eur. Phys. J. D **48**, 127 (2008)
115. K. Baumann, C. Guerlin, F. Brennecke, T. Esslinger, Nature (London) **464**, 1301 (2010)

116. J. Klinder, H. Keßler, M. Bakhtiari, M.T. Reza, A. Hemmerich, Phys. Rev. Lett. **115**, 230403 (2015)
117. F. Piazza, H. Ritsch, Phys. Rev. Lett. **115**, 163601 (2015)
118. H. Kessler, J.G. Cosme, M. Hemmerling, L. Mathey, A. Hemmerich, Phys. Rev. A **99**, 053605 (2019)
119. P. Zupancic, D. Dreon, X. Li, A. Baumgärtner, A. Morales, W. Zheng, N.R. Cooper, T. Esslinger, T. Donner, Phys. Rev. Lett. **123**, 233601 (2019)
120. M. Greiner, O. Mandel, T. Esslinger, T.W. Hänsch, I. Bloch, Nature **415**, 39 (2002)
121. Y. Kato, Q. Zhou, N. Kawashima, N. Trivedi, Nat. Phys. **4**, 617 (2008)
122. M.P.A. Fisher, P.B. Weichman, G. Grinstein, D.S. Fisher, Phys. Rev. B **1**, 546 (1989)
123. F. Serwane, G. Zürn, T. Lompe, T. Ottenstein, A. Wenz, S. Jochim, Science **332**, 336 (2011)
124. S. Murmann, A. Bergschneider, V.M. Klinkhamer, G. Zürn, T. Lompe, S. Jochim, Phys. Rev. Lett. **114**, 080402 (2015)
125. U. Schollwöck, Rev. Mod. Phys. **77**, 259 (2005)
126. J. Dobrzyniecki, T. Sowiński, Phys. Lett. A **382**, 394 (2018)
127. J. Dobrzyniecki, X. Li, A.E. Nielsen, T. Sowiński, Phys. Rev. A **97**, 013609 (2018)
128. E. Fasshauer, A.U.J. Lode, Phys. Rev. A **93**, 033635 (2016)
129. A.U.J. Lode, C. Bruder, Phys. Rev. A **94**, 013616 (2016)
130. D. Murphy, J. McCann, J. Goold, T. Busch, Phys. Rev. A **76**, 053616 (2007)
131. D. Murphy, J. McCann, Phys. Rev. A **77**, 063413 (2008)
132. J.R. Salgueiro, M. Zacarés, H. Michinel, A. Ferrando, Phys. Rev. A **79**, 033625 (2009)

Atomistic Simulations of the Human Proteasome Inhibited by a Covalent Ligand

Michal H. Kolář, Lars V. Bock, and Helmut Grubmüller

Abstract The proteasome is a large biomolecular complex responsible for protein degradation. It is under intense research due to its fundamental role in cellular homeostasis, and tremendous potential for medicinal applications. Recent data from X-ray crystallography and cryo-electron microscopy have suggested that there is a large-scale structural change upon binding of an inhibitor. We carried out atomistic molecular dynamics simulations of the native and inhibited proteasomes to understand the molecular details of the inhibition. Here we describe the technical details of the simulations and assess the quality of the trajectories obtained. The biochemical aspects of the proteasome are under further investigation and will be published elsewhere. This work was a part of the GSC-Prot project at the HLRS, run on the Cray XC40 supercomputing system.

1 Introduction

Over the many years since the pioneering studies [1, 2], biomolecular molecular dynamics (MD) simulations have become a valuable source of scientific data. They capture functional motions of biomolecules with high spatial and temporal resolution and bring information about dynamics and energetics. They complement classic biophysical techniques for structure determination [3, 4], facilitate drug design [5] or successfully tackle important questions of molecular biology [6–8].

Routinely, microsecond-long trajectories of systems up to few tens of thousands atoms can be achieved on workstations equipped by the customer-class graphical processor units [9]. However, simulating larger assemblies and multi-component biomolecular complexes, such as ribosome [10] or proteasome [11], remains a challenge, and can only be done on high-performance supercomputers [12], or through distributed computing [13, 14].

M. H. Kolář (✉) · L. V. Bock · H. Grubmüller
Max Planck Institute for Biophysical Chemistry, Am Fassberg 11, 37077 Göttingen, Germany
e-mail: michal@mhko.science

M. H. Kolář
University of Chemistry and Technology, Technicka 5, 16628 Prague, Czech Republic

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_3

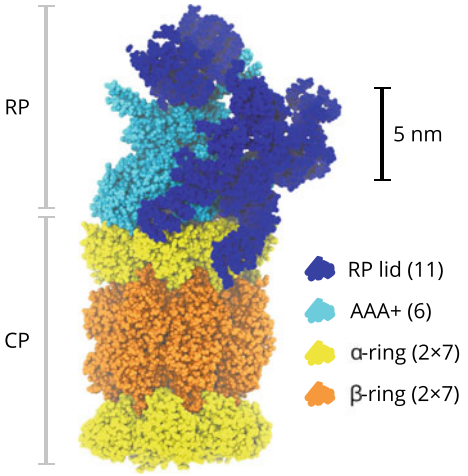


Fig. 1 Anatomy of the human proteasome. In the legend, numbers of protein subunits are given in parentheses. Prepared from PDB 5m32 [17].

Here, we present results of atomistic MD simulations of human proteasome on a multi-microsecond time scale. The proteasome is a stochastic 2.5 MDa nanomachine responsible for protein degradation in eukaryotic cells via the ubiquitin-proteasome pathway [15]. It helps maintaining the delicate balance of protein concentrations, and thus plays a fundamental role in cell life cycle. Modulation of proteasome function has a direct effect on cell homeostasis, disruption of which often leads to the cell death [16].

The proteasome contains two major functional parts: a 20S core particle (CP) and a 19S regulatory particle (RP), which together form the 26S particle depicted in Fig. 1. The CP is a barrel-shape complex of several protein subunits organized in four rings - two α-rings and two β-rings in a stacked $\alpha\beta\beta\alpha$ arrangement. Three β-subunits (β_1, β_2, and β_5) have been shown to catalyze the proteolysis. The RP consists of a base and a lid. The base is formed by a ring of six distinct ATPases associated with diverse cellular activities (AAA+), so called regulatory particle triple A proteins. The AAA+ works as an engine that pushes the substrate into the CP [18]. In addition, several non-ATPase subunits belong to the base and are involved in the recognition of proteasome substrates. Overall, the RP lid consists of eleven different subunits, which recognize and pre-processes protein substrate before it is transferred into CP for degradation.

Large amounts of structural information have been gathered since the proteasome discovery. Recently, several groups have determined the 26S proteasome structure at atomic or near-atomic resolution [17, 19–21]. Despite the continuous efforts, some parts of RP still remain unresolved, mostly due to their high inherent mobility. Understanding the proteasome structure and function poses a fundamental scientific challenge. However, the proteasome is under intensive investigation also due to a tremendous potential for medicinal applications [22, 23].

Oprozomib (OPR) [24] is one of the ligands which have already reached the market as potent anti-cancer agents. The X-ray crystal structure of the CP-OPR complex,

resolved at resolution of 1.8 Å, revealed two OPR molecules in the CP – one per each of the two β_5-subunits. Remarkably, the inhibited CP structure is highly similar to the structure of native CP (at 1.9 Å) [25]. The protein backbone root-mean-square deviation (RMSD) is only 0.4 Å, hence the inhibitors do not induce marked structural changes of the CP.

However, based on cryo-electron microscopy (cryo-EM) structures of the 26S proteasome, it has been recently suggested that the RP undergoes a large conformational change upon inhibition [17]. When OPR is bound to the CP β_5-subunits, the RP rotates by about 25° into a non-productive state. Intriguingly, the drug binding triggers an allosteric signal which is transferred and amplified over a distance larger than 150 Å, while keeping the (average) structure of the CP almost intact. Our main objective is to understand the atomic details of the CP-RP mutual motion possibly triggered by the OPR binding.

Here we present the computational details of our MD simulations performed on HLRS Hazel Hen, and assess the quality of the trajectories obtained.

2 Methods

2.1 Simulated Systems

We have built four proteasome constructs: native CP, inhibited CP, native AA, and inhibited AA, where AA stands for a CP with one AAA+ ring (Fig. 2). The inhibited AA structure was prepared from the available cryo-EM data (PDB 5m32 [17]). The dangling N-terminal α-helices of the AAA+ regulatory subunits 6A, 6B, 8 and 10 (UniProt naming convention) were omitted. Where needed, the CP subunits were completed by missing amino acids to keep the up-down sequence symmetry of the pairs of α- and β-rings. By removing the two inhibitors from the inhibited AA, we prepared the native AA structure. Experimentally determined coordinates of water molecules, K^+, Mg^{2+} and Cl^- ions were taken from the X-ray data (PDB 5le5 [25]) and added after superimposing the backbone atoms of the CP. Each of the six AAA+ subunits contained one adenosine diphosphate as found in the cryo-EM model.

Each construct was placed into a periodic rhombic dodecahedron box of sufficient size such that the distance between the solute and box faces was not shorter than 1.5 nm. The system was dissolved in a solution of K^+ and Na^+ ions of the excess concentrations of 139 mM and 12 mM, respectively, and neutralized by Cl^- anions. In total, the simulations contained about 0.8 and 1.6 million atoms, for the CP and AA constructs, respectively.

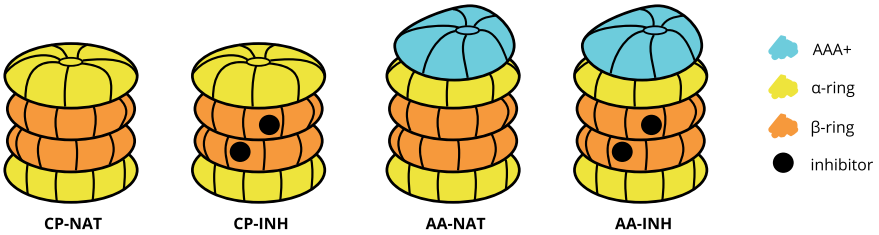


Fig. 2 A scheme of the four proteasome constructs simulated.

2.2 Simulation Setup

Molecular dynamics simulations were carried out using classical interaction potentials. A standard version of the Amber force field was used for the proteasome [26]. The OPR was parametrized together with its covalently bound N-terminal threonine in the spirit of the Amber family of force fields, using fitted partial atomic charges from the Restricted Electrostatic Potential method [27], and General Amber Force Field parameters [28]. The TIP3P water model [29] and ion parameters by Joung and Cheatham [30] were used.

Newton's equations of motion were integrated using the leap-frog algorithm. All bonds were constrained to their equilibrium lengths using the parallel LINCS algorithm of the sixth order [31]. Hydrogen atoms were converted into virtual sites [32] which allowed using 4-fs integration time step in the production simulations. Electrostatic interactions were treated by the Particle Mesh Ewald method [33] with the direct space cut-off of 1.0 nm and 0.12 nm grid spacing. Van der Waals interactions, described by the Lennard-Jones potential, used a cut-off of 1.0 nm.

The systems were equilibrated in several steps. First, each system was thoroughly energy minimized. In some instances, the virtual-site model caused crashes, thus for the minimization a model with explicit hydrogen atoms and flexible water molecules was used. Second after minimizing water molecules in roughly 50,000 steps of the steepest descent algorithm, the water was heated from 10 K to 300 K in a 5-ns long constant-volume MD simulation, where two thermostats were used separately for the solute and solvent. Velocities were selected randomly from the Maxwell-Boltzmann distribution at the given temperature. Moreover during heating, the solute heavy atoms were restrained by harmonic potential to their starting coordinates with the force constant of 5000 kJ mol^{-1} nm^{-2}. Next, the density of the system was equilibrated in a 20-ns long constant-pressure MD simulation at 300 K and 1 bar, where v-rescale thermostat [34] and Berendsen barostat [35] were used. During this step, the solute was still restrained. Finally, the position restraints were gradually released in a 50 ns-long simulation, where the force constant was interpolated between its initial value and zero.

In production runs, the isobaric-isothermal statistical ensembles were generated at 300 K and 1 bar using the v-rescale thermostat and Parrinello-Rahman barostat

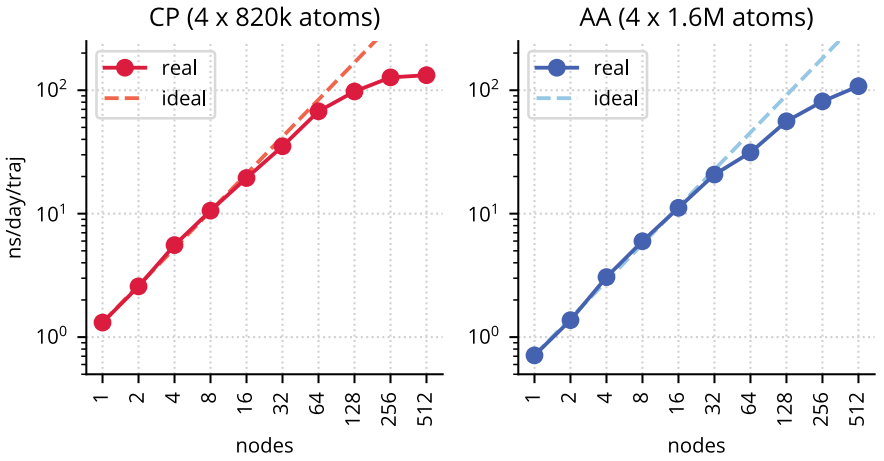


Fig. 3 Performance of GROMACS 2016 on the Hazel Hen supercomputer.

[36], respectively. We simulated roughly 4 μs per trajectory with the exception of the CP-INH, where the length was increased up to 5.6 μs. For validation of the observed phenomena, another set of simulations was carried out. This started from the final conformation of the inhibited proteasome simulations, where the inhibitor was removed. Due to technical reasons, only the coordinates of the solute were kept, whereas the water and ions were added from scratch in the same manner as with the simulations initiated from the experimental conformation.

2.3 Software Details

The simulations were carried out in the GROMACS 2016 package [37]. It is a well-established, highly-optimized C/C++ code released under Lesser General Public License. Initially, we used the standard module available on HLRS Hazel Hen supercomputer. After removal of version 2016 from the list of supported modules, we used a self-compiled version with very similar performance characteristics.

GROMACS uses a mixed MPI/OpenMP parallelization which may scale down to "few tens of atoms per core" [37]. In our case, the scaling was better for the larger AA than smaller CP construct (Fig. 3). For the production runs, we employed 128 or 256 nodes with two 12-core Intel Xeon (Haswell gen.) processors each. For each system, four independent trajectories initiated with different velocities from the Maxwell-Boltzmann distribution were generated. To improve scaling, bundles of the four simulations were run as a single *aprun* argument.

Memory requirements were rather low, our system consumed about 600 MB of memory per MPI task (for the AA construct) comprising several OpenMP threads.

The simulations generated large amounts of data compressed to a high degree at the level of GROMACS output routines. GROMACS allows checkpoint-file dependent restarts of the simulations so the simulations were run as chained jobs to fit the maximum runtime of 24 h on Hazel Hen. Due to our interest in the solute behavior and limited disk space, we saved the water coordinates less frequently (100 ps) than the coordinates of the solute (10 ps). Each chain step produced a portion of solute trajectory of about 5.5 GB. Due to the limited disk space, these portions were downloaded frequently to our local servers, and concatenated before the final analysis.

3 Results and Discussion

For all trajectories, we calculated the root-mean-square deviations (RMSDs) of the backbone atoms with respect to the starting proteasome conformation according to Eq. 1. The analysis was performed after a least-square alignment of the trajectory to the starting conformation using the backbone atoms of the CP subunits.

$$\mathrm{RMSD}(t) = \sqrt{\frac{1}{N_a}\sum_{a}^{N_a}(\mathbf{r}_a(t) - \mathbf{r}_a(0))^2}, \tag{1}$$

where N_a is the number of atoms in a trajectory, the $\mathbf{r}_a(0)$ is position vector at time 0, i.e. the experimental structure, and $\mathbf{r}_a(t)$ is the position vector at time t.

All of the trajectories appear stable within the limits of such a simple measure as RMSD. Figure 4 shows the RMSD profiles with the averages over respective trajectories between 0.30 and 0.35 nm. These values are expected, given the size of the system (over 6,000 amino acids) and no significant drift. Similar plots for AA constructs are in Fig. 5. Here the RMSD values averaged over the trajectories are around 0.40 nm, with two instances higher than 0.45 nm. The profiles show no significant drift. Higher RMSD values are related to the size of the system (over 8300 amino acids), and to the fact that only the CP subunits were used for the alignment.

The natural sequence symmetry of the CP allowed us to assess the convergence of the simulations. The CP is free to move in the simulation box, so the structure and dynamics of the subunits in the upper and lower halves should converge to common values if the free-energy minimum is well defined.

For the CP constructs, we calculated the average conformation between 1600 and 3600 ns of each trajectory, i.e. the mean position vector x of all heavy atoms. We aligned the upper and lower halves using the backbone atoms. Then for each pair of equivalent subunits in the upper and lower half, the $\mathrm{RMSD}_{\mathrm{u-l}}$ was calculated as follows.

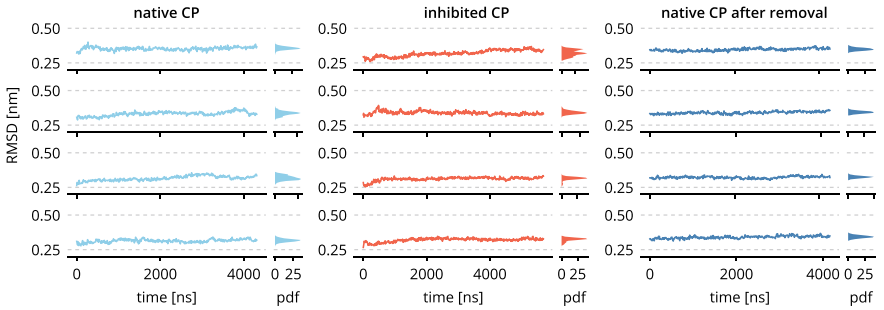


Fig. 4 Time evolution of the backbone root-mean-square deviation (RMSD) and the respective probability density functions (pdf) obtained for the CP constructs. Four independent trajectories of the inhibited constructs are shown in red, the native in blue. The pale blue traces started from the experimental conformation, whereas the dark blue started from the final conformations of the inhibited constructs. (Color figure online)

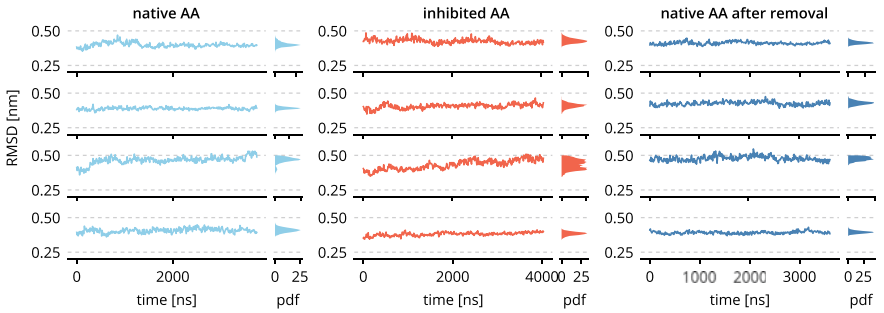


Fig. 5 Time evolution of the backbone root-mean-square deviation (RMSD) and the respective probability density functions (pdf) obtained for the AA constructs. Four independent trajectories of the inhibited constructs are shown in red, the native in blue. The pale blue traces started from the experimental conformation, whereas the dark blue started from the final conformations of the inhibited constructs. (Color figure online)

$$\mathrm{RMSD}_{\mathrm{u-l}} = \sqrt{\frac{1}{N_a}\sum_{a}^{N_a}\left(\mathbf{r}_{a,u} - \mathbf{r}_{a,l}\right)^2}, \tag{2}$$

where $\mathbf{r}_{a,u}$ is the position vector of atom a in the upper half, the $\mathbf{r}_{a,l}$ is the position vector of the equivalent of atom a in the lower half, and the sum runs over N_a atoms on one proteasome half. If the mean structures were identical as proposed by the sequence symmetry, the $\mathrm{RMSD}_{\mathrm{u-l}}$ would be zero. Non-zero values indicate structural variation between equivalent subunits.

Figure 6 shows the $\mathrm{RMSD}_{\mathrm{u-l}}$ of CP constructs obtained from the experimental structure and from the simulations. The non-zero values in the experimental structures may be related to crystal-packing effects. Moreover, there are number of surface protein loops and terminals which are flexible. Thus, the free-energy surface is expected

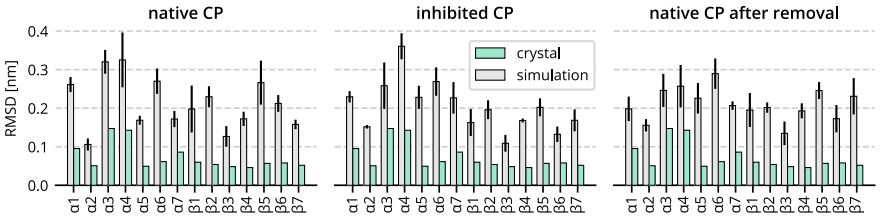


Fig. 6 The $RMSD_{u-l}$ of the CP constructs. Values from the crystal structure are shown in green. The gray bars represent error bars represent standard errors of the mean obtained from four independent trajectories. (Color figure online)

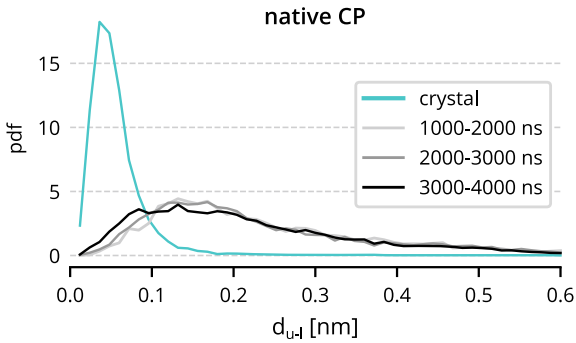


Fig. 7 Histogram of distances between equivalent backbone atoms of the upper and lower halves in the crystal structure (green) and simulation (gray). Data from various parts of one trajectories are shown in shades of gray. (Color figure online)

to feature many shallow minima, so their conformation in the upper and lower halves may vary. The $RMSD_{u-l}$ values from the simulations are higher, for β-units by factor of about 3, for α-units by factor of about 2.

Further after the least-square alignment of the backbone atoms of the two proteasome halves, we calculated distances d_{u-l} between equivalent atoms in upper and lower halves. The histogram of d_{u-l} shows (Fig. 7) a maximum about 0.05 nm for the crystal. For simulation, the maximum lies slightly beyond 0.1 nm and is broader. This indicates that the proteasome conformations averaged over a trajectory are structurally less symmetric than the crystal. A projection of the simulation d_{u-l} onto the proteasome structure (Fig. 8) reveals that the largest structural variations are located in the surface loops and terminal chains. In the course of simulation time, the d_{u-l} profiles do not diverge (Fig. 7), or even slightly improve towards shorter values.

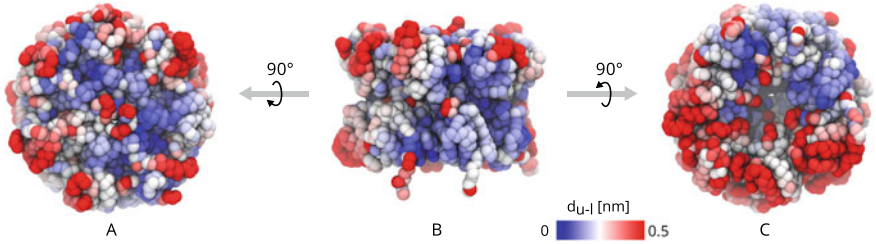


Fig. 8 Projection of the distance d_{u-l} between equivalent atoms of the upper and lower halves onto the crystal structure of the proteasome oriented towards the α-ring (A), front face (B) and β-ring (C). Each sphere represents one amino acid and the color scale goes from blue (low d_{u-l}) through white to red (high d_{u-l}). (Color figure online)

4 Concluding Remarks

Using the Cray XC40 supercomputer, we have performed atomistic MD simulations of the proteasome, a multi-protein complex responsible for protein degradation recently used as an anti-cancer drug target. We obtained trajectories totalling 100 μs in length of several systems with 0.8 and 1.6 million atoms.

Here, we have presented a technical report focused on the simulation setup, runtime performance and basic analyses. Next, we will focus on proteasome function and its regulation and will present such biochemical aspects in future texts. The trajectories obtained through the MD simulations are likely of sufficient quality to explain at atomic level what changes the inhibitor OPR triggers and how these can modulate proteasome function.

References

1. M. Levitt, A. Warshel, Computer simulation of protein folding. Nature **253**, 694–698 (1975)
2. J.A. McCammon, B.R. Gelin, M. Karplus, Dynamics of folded proteins. Nature **267**, 585–590 (1977)
3. L.G. Trabuco, E. Villa, K. Mitra, J. Frank, K. Schulten, Flexible fitting of atomic structures into electron microscopy maps using molecular dynamics. Structure **16**, 673–683 (2008)
4. M. Igaev, C. Kutzner, L.V. Bock, A.C. Vaiana, H. Grubmüller, Automated Cryo-EM structure refinement using correlation-driven molecular dynamics. eLife **8**, e43542 (2018)
5. P. Śledź, A. Caflisch, Protein structure-based drug design: from docking to molecular dynamics. Curr. Opin. Struct. Biol. **48**, 93–102 (2018)
6. K. Lindorff-Larsen, S. Piana, R.O. Dror, D.E. Shaw, How fast-folding proteins fold. Science **334**, 517–520 (2011)
7. K.J. Kohlhoff et al., Cloud-based simulations on Google exacycle reveal ligand modulation of GPCR activation pathways. Nat. Chem. **6**, 15–21 (2014)
8. W. Kopec et al., Direct knock-on of desolvated ions governs strict ion selectivity in K + channels. Nat. Chem. **10**, 813–820 (2018)

9. C. Kutzner, S. Páll, M. Fechner, A. Esztermann, B.L. de Groot, H. Grubmüller, More bang for your buck: improved use of GPU nodes for GROMACS 2018. J. Comput. Chem. **40**, 2418–2431 (2019)
10. L.V. Bock, M.H. Kolář, H. Grubmüller, Molecular simulations of the ribosome and associated translation factors. Curr. Opin. Struct. Biol. **49**, 27–35 (2018)
11. M. Wehmer et al., Structural insights into the functional cycle of the ATPase module of the 26S proteasome. Proc. Natl. Acad. Sci. **114**, 1305–1310 (2017)
12. W.E. Nagel, D.H. Kröner, M.M. Resch, *High Performance Computing in Science and Engineering'18: Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2018* (Springer, Heidelberg, 2019)
13. T.J. Lane, D. Shukla, K.A. Beauchamp, V.S. Pande, To milliseconds and beyond: challenges in the simulation of protein folding. Curr. Opin. Struct. Biol. **23**, 58–65 (2013)
14. S. Chen et al., The dynamic conformational landscape of the protein methyltransferase SETD8. eLife **8**, e45403 (2019)
15. Y. Saeki, K. Tanaka, in *Ubiquitin Family Modifiers and the Proteasome*, ed. by R.J. Dohmen, M. Scheffner. Methods in Molecular Biology, vol. 832 (Humana Press, Totowa, 2012), pp. 315–337
16. M. Rechsteiner, C.P. Hill, Mobilizing the proteolytic machine: cell biological roles of proteasome activators and inhibitors. Trends Cell Biol. **15**, 27–33 (2005)
17. D. Haselbach, J. Schrader, F. Lambrecht, F. Henneberg, A. Chari, H. Stark, Long-range allosteric regulation of the human 26S proteasome by 20S proteasome-targeting cancer drugs. Nat. Commun. **8**, 1–8 (2017)
18. S. Bar-Nun, M.H. Glickman, Proteasomal AAA-ATPases: structure and function. Biochimica et Biophysica Acta (BBA) - Mol. Cell Res. **1823**, 67–82 (2012)
19. A. Schweitzer et al., Structure of the human 26S proteasome at a resolution of 3.9 Å. Proc. Natl. Acad. Sci. **113**, 7816–7821 (2016)
20. X. Huang, B. Luan, J. Wu, Y. Shi, An atomic structure of the human 26S proteasome. Nat. Struct. Mol. Biol. **23**, 778–785 (2016)
21. S. Chen et al., Structural basis for dynamic regulation of the human 26S proteasome. Proc. Natl. Acad. Sci. **113**, 12991–12996 (2016)
22. L. Bedford, J. Lowe, L.R. Dick, R.J. Mayer, J.E. Brownell, Ubiquitin-like protein conjugation and the ubiquitin-proteasome system as drug targets. Nat. Rev. Drug Discov. **10**, 29–46 (2011)
23. A.F. Kisselev, W.A. van der Linden, H.S. Overkleeft, Proteasome inhibitors: an expanding army attacking a unique target. Chem. Biol. **19**, 99–115 (2012)
24. S.D. Demo et al., Antitumor activity of PR-171, a novel irreversible inhibitor of the proteasome. Can. Res. **67**, 6383–6391 (2007)
25. J. Schrader et al., The inhibition mechanism of human 20S proteasomes enables next-generation inhibitor design. Science **353**, 594–598 (2016)
26. K. Lindorff-Larsen et al., Improved side-chain torsion potentials for the Amber ff99SB protein force field. Proteins Struct. Funct. Bioinform. **78**, 1950–1958 (2010)
27. C.I. Bayly, P. Cieplak, W. Cornell, P.A. Kollman, A well-behaved electrostatic potential based method using charge restraints for deriving atomic charges: the RESP model. J. Phys. Chem. **97**, 10269–10280 (1993)
28. J. Wang, R.M. Wolf, J.W. Caldwell, P.A. Kollman, D.A. Case, Development and testing of a general amber force field. J. Comput. Chem. **25**, 1157–1174 (2004)
29. W.L. Jorgensen, J. Chandrasekhar, J.D. Madura, R.W. Impey, M.L. Klein, Comparison of simple potential functions for simulating liquid water. J. Chem. Phys. **79**, 926–935 (1983)
30. I.S. Joung, T.E. Cheatham, Determination of alkali and halide monovalent ion parameters for use in explicitly solvated biomolecular simulations. J. Phys. Chem. B **112**, 9020–9041 (2008)
31. B. Hess, P-LINCS: a parallel linear constraint solver for molecular simulation. J. Chem. Theory Comput. **4**, 116–122 (2008)
32. H.J.C. Berendsen, W.F. Van Gunsteren, in *Molecular Liquids: Dynamics and Interactions*, ed. by A.J. Barnes, W.J. Orville-Thomas, J. Yarwood. NATO ASI Series (Springer, Dordrecht 1984) pp. 475–500

33. T. Darden, D. York, L. Pedersen, Particle mesh Ewald: an N·log(N) method for Ewald sums in large systems. J. Chem. Phys. **98**, 10089–10092 (1993)
34. G. Bussi, D. Donadio, M. Parrinello, Canonical sampling through velocity rescaling. J. Chem. Phys. **126**, 014101 (2007)
35. H.J.C. Berendsen, J.P.M. Postma, W.F. van Gunsteren, A. DiNola, J.R. Haak, Molecular dynamics with coupling to an external bath. J. Chem. Phys. **81**, 3684–3690 (1984)
36. M. Parrinello, A. Rahman, Polymorphic transitions in single crystals: a new molecular dynamics method. J. Appl. Phys. **52**, 7182–7190 (1981)
37. M.J. Abraham et al., GROMACS: high performance molecular simulations through multi-level parallelism from laptops to supercomputers. SoftwareX **1–2**, 19–25 (2015)

Accreting Black Hole Binaries

Roman Gold, Luciano Rezzolla, Ludwig Jens Papenfort, Samuel Tootle, Hector Olivares, Elias Most, Ziri Younsi, Yosuke Mizuno, Oliver Porth, and Christian Fromm

Abstract We describe our utilization of HLRS resources in 2020. Chief among this has been a generation of complete spacetime initial data and a completion of a seamless interface between the BHAC code and the Kadath library. We can now access Kadath routines directly from within BHAC crucially avoiding unnecessary IO operations and enabling a spectrally accurate evaluation of metric quantities and its derivatives on any grid (static or dynamic). These developments form a crucial pillar for continued progress including follow-up work even after the BBHDISKS project finishes. Second, the numerical treatment of the sensitive horizon region has been successfully completed and is capable to handle evolutions on par with single black hole evolutions. Additionally we have completed tests of some of our algorithms that directly benefit the funded project. We have refined GRMHD libraries for an improved interpretation of the first black hole image. Lastly, we have started preparatory work to set up a pipeline to convert our novel accreting black-hole binary simulations into images and movies. The BHOSS code is now capable of producing images in black hole binary spacetimes and remaining development tasks to fully connect it to the matter evolutions obtained in this project are now modest.

Research field:
General relativity/Astrophysics

1 Introduction

The allocation "44149 BBHDISKS" on the system CRAY XC40 (HAZEL HEN) at HLRS has been awarded to our research group in August 2018 and extended until August 2020 and presents us with a significant computational resource to carry out this project.

R. Gold (✉) · L. Rezzolla · L. J. Papenfort · S. Tootle · H. Olivares · E. Most · Z. Younsi · Y. Mizuno · O. Porth · C. Fromm
Institute for Theoretical Physics (ITP), Goethe University, Max-von-Laue-Str. 1, 60438 Frankfurt am Main, Germany
e-mail: gold@sdu.dk

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_4

The scientific research carried out in the group is broad, covering a number of active topics in the field of general relativistic astrophysics and compact objects most of which require substantial computational resources. Specifically these topics include:

- the general behaviour of magnetized, turbulent gas moving around or falling onto a black hole systems
- the observational appearance of said systems
- the predictions from alternative theories of gravity
- the predictions for systems with two black holes in magnetized gas
- advancing our computational tools to tackle the above systems in the contexts of both ideal and resistive magnetohydrodynamics on non-homogenous and dynamical grids.

Our studies exclusively probe systems governed by coupled, non-linear partial differential equations of Magnetohydrodynamics in strong-field gravity both GR and non-GR. In our research we therefore heavily rely on numerical methods and large scale computational facilities such as the ones provided by HLRS, which are a very important asset to us.

Due to various circumstances beyond our control such as the shutdown of Hazelhen and problems with setting up Hawk related to a cybersecurity incident we have not been able to make use of our allocation in a crucial time of our project and had to rely on other resources to execute our simulations on a smaller scale. This has been a risk that we were prepared for as detailed in our proposal. A completion of our research program as outlined does, however, rely on the utilization of Hawk and we will start migrating our software over to Hawk once the system comes online. While we were able to identify and cure more obvious and small bugs in our codes, we will need to run a few last full tests on Hawk for both reliability and performance before entering full production.

2 Numerical Methods

The simulations are performed using the recently developed code BHAC [15], which is based on the public `MPI-AMRVAC` Toolkit [16]. BHAC was built to solve the general-relativistic MHD equations in the ideal limit (i.e., for plasmas with infinite conductivity) on spacetimes that are fixed but arbitrary and can therefore model stationary BH spacetimes in any theory of gravity [11, 17]. The code is currently used within the Event Horizon Telescope collaboration, and in particular, the ERC synergy grant "BlackHoleCam" [8] to model mm-VLBI observations of the supermassive BH M87* [2–7] and the candidate at the galactic centre Sgr A* to simulate plasma flows onto compact objects [12, 15]. BHAC offers a variety of numerical methods, fully adaptive block based (oct-) tree mesh refinement with a fixed refinement factor of two between successive levels, and is already interfaced with many analysis tools or

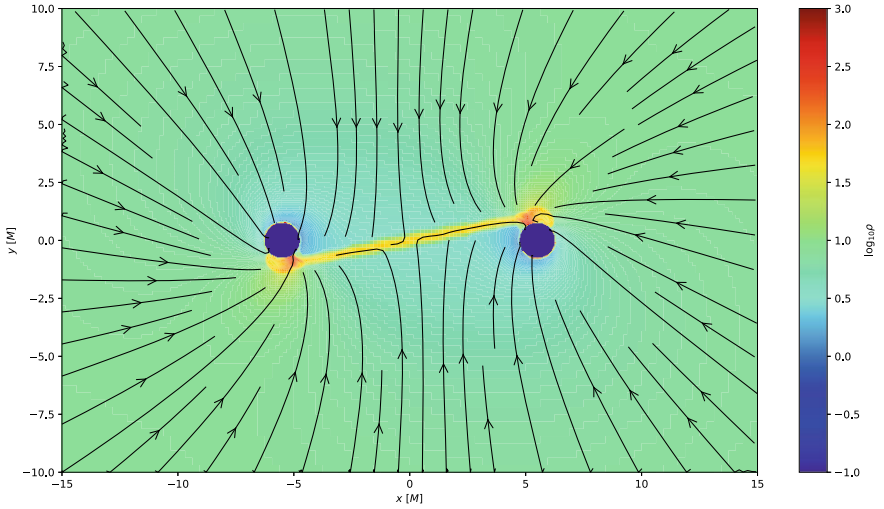


Fig. 1 Central rest-mass density (colour scale) of a gas cloud corotating with a black hole binary evolved with BHAC and the four velocity (arrows). One can clearly see the black hole horizons (dark blue), a pile-up of gas in front of the orbit of each black hole, a trailing wake of lowered density, and a dense stream of gas between the black holes reminiscent of equipotential surfaces in the tidal field of the binary.

external codes used by our group. In all the above studies so far the spacetime metric is analytically given and manifestly stationary.

A key requisite to evolve accreting black hole binary systems is the generation of a valid spacetime metric featuring two black holes orbiting around each other that satisfies Einstein's field equations of General Relativity. Such initial conditions cannot be found by simply superimposing two known black hole solutions as would be possible in Newtonian gravity. Instead, rather complex non-linear elliptic equations have to be solved at very high accuracy. Another challenge is very similar but even more challenging than for single black holes: The initial conditions for the matter are found from initially well-defined but not realistic equilibrium condition, that are evolved in time until the matter dynamically adjusts to the effects of the magnetic fields and binary spacetime.

3 Ongoing Accreting Binary Black Holes Project

In the current project BBHDISKS we have now achieved matter evolutions in spacetime metrics of two black holes in orbit around each other, see e.g. Figures 1 and 2. Thanks to a novel and innovative approach we were able to exploit both symmetries in the spacetime (preserve the corotating frame and spectral accuracy) as well as key advantages of the full AMR, state-of-the-art code BHAC. Both of these features are a first and present a fundamental and substantial advantage over any other study in this field.

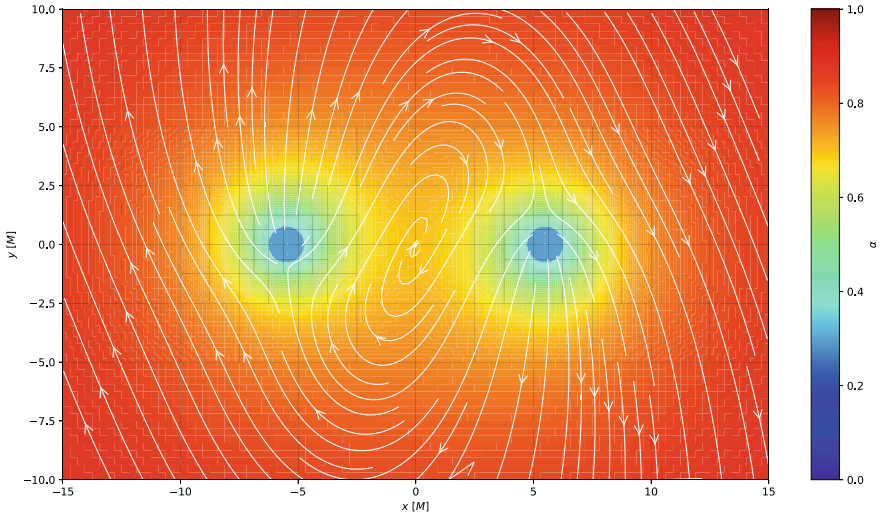


Fig. 2 Lapse metric function (color scale) and shift vector (arrows) in the equatorial plane of the irrotational binary black hole system. The spacetime is a full solution to Einstein's equations and is mapped onto an adaptive mesh refinement grid in BHAC (each grid block represents 16 grid cells for visual clarity).

The first non-trivial step was to obtain valid and spectrally accurate (i.e. solving Einstein's field equations) initial data. Binary BH initial data are being computed using `Kadath` [10], which works within the conformal thin-sandwich approximation (CTS). In CTS it is assumed that the binary spacetime has a helical Killing vector (i.e., the binary is stationary in a co-rotating frame) and is conformally flat [13].

The second step that we have achieved is that we have now linked BHAC to the `Kadath` library [10] enabling a flexible, convenient, and efficient interface between the spacetime metric solver and `BHAC`. With this coupling in place we can now evaluate the spacetime metric to spectral accuracy on any (e.g. non-uniform) numerical grid without any tedious and unnecessary I/O operations.

Third, we have successfully initialized and evolved matter configurations within a binary spacetime. Specifically, we were able to make the necessary adjustments to handle the more complicated nature of the black hole horizons.

4 Job Statistics and Resources Usage

The typical job size for our projects is largely dependent on the type of physical system we are modelling, the microphysical processes involved and of course the level of accuracy needed, which sets constraints on numerical parameters such as the resolution, domain size and order of the method. Nonetheless we can provide some

estimates. A typical job can require the use of 50 nodes (1200 cores) for a period of 24 h at low resolution and twice as much (100 nodes/2400 cores) for several days at mid-high resolution. In the hybrid MPI/OpenMP parallelization scheme that we employ, this results in a total of 50 to 100 MPI processes (2 per node), each of them spawning 12 threads each. We normally do not exploit hyperthreading and attempt to bind each thread to a separate physical core.

A grand total of 51473420 RTh were awarded to us on HAWK, and we have used 25472 RTh (6.3%) and 2694264 (5.2%) RTh on Hazelhen at this point into our allocation as of 15/06/2020. The shutdown of Hazelhen and problems in setting up Hawk related to the cybersecurity incident have prevented us from using the allocation in an important point into the project. Instead, we resorted to smaller test simulations on local resources where also faster iterations between runs is possible. However a few full scale tests have to be run on Hawk once it becomes available and we have ported out siftware over successfully.

At present 5 researchers within our group have access to computing resources at HLRS.

We have participated three times in the HLRS code optimisation workshop (recently in November) to achieve optimal performance on the Cray XC40 system and plan to do so again in the course of this allocation when transitioning to the new Hawk system.

4.1 Refining the Library of GRMHD Images for the EHT

Three-dimensional general relativistic magnetohydrodynamic simulations of accretion onto supermassive black holes were performed at the HazelHen supercomputer using the code BHAC, as part of the efforts to build a library of simulations and images to aid in the interpretation of the first horizon-scale radio images of a supermassive black hole candidate [2–7].

We are currently refining the library of GRMHD-based images for a more in depth comparison to observational data. Key improvements are a better coverage of the underlying parameter space and higher image resolution.

4.2 Preparatory Development on Ray Tracing in Binary Spacetimes

While our production phase was delayed due to the shutdown of Hazelhen and the delay in getting started on Hawk we have instead started other development earlier than original envisaged. Once full production is underway we seek to form synthetic images as they would appear to a distant observer. Such images require the solution of a synchrotron radiative transfer problem along the trajectories of light rays in

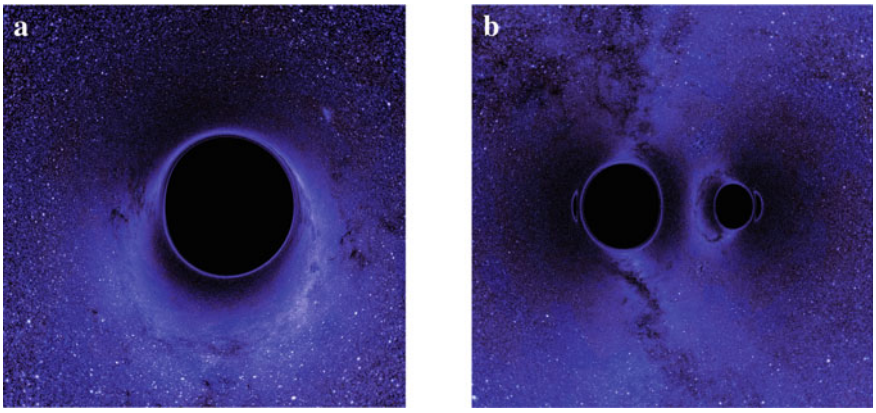


Fig. 3 Ray-traced images computed with the BHOSS code [18] through a single (left panel) and binary (right panel) black hole spacetime. Note the peculiar lensing effects in the binary case that are absent in the single black hole case.

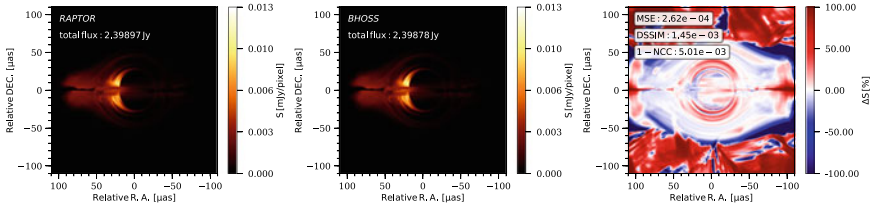


Fig. 4 Comparisons of ray-traced images computed with the BHOSS code [18] and RAPTOR [1]

the underlying spacetime. For single black holes there are many codes capable of doing this and many are used within the EHT, see Sect. 5.3. When it comes to binary spacetimes the situation is very different. The BHOSS code is now capable of tracing light paths in binary spacetimes, see Fig. 3 as an example.

5 Completed Projects and Publications

5.1 Completion of the Library of GRMHD Images for the EHT

As discussed in Sect. 4.1, the construction of the GRMHD image library was successfully completed and extensively used in the theoretical interpretations of EHT results.

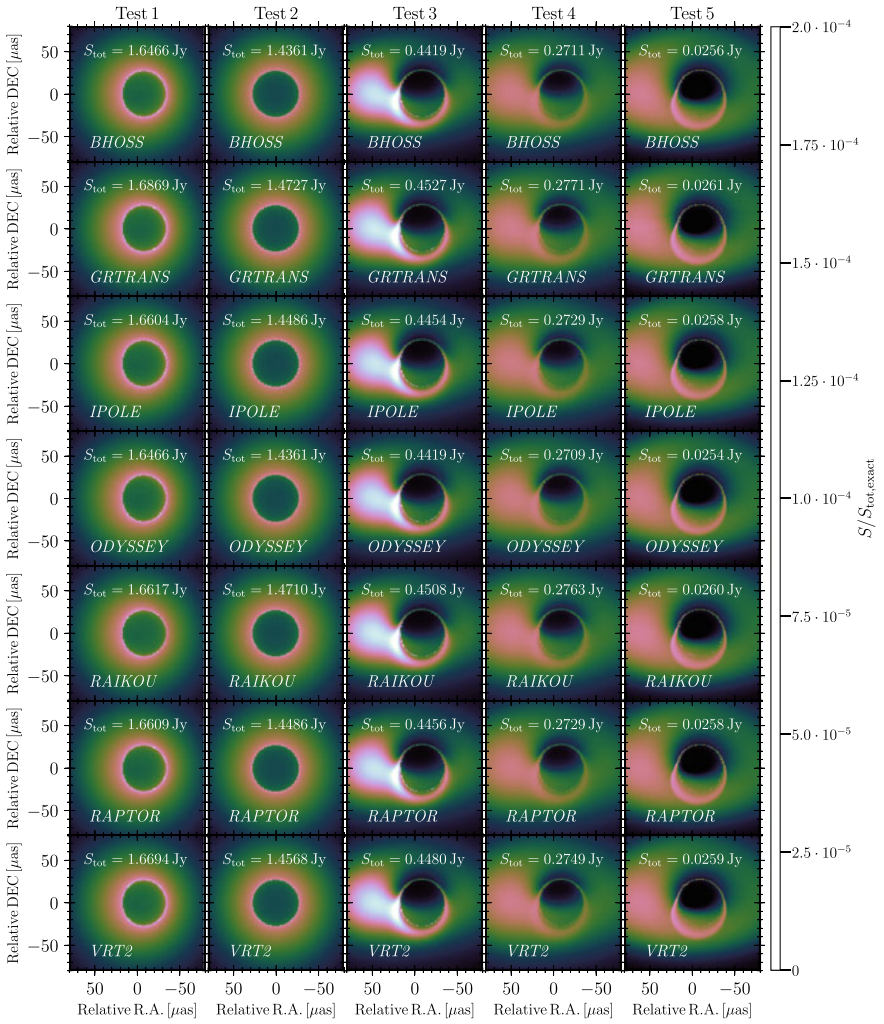


Fig. 5 Comparisons of ray-traced images across all codes used within the EHT

5.2 GRMHD Code Comparison Project

In [14] we have published the results of an exhaustive comparison of various independent GRMHD codes in an effort to quantify the degree of agreement among different codes when evolving the same initial conditions. The main difficulty is that the underlying dynamics features turbulent motion that is exponentially sensitive to the initial data and therefore cannot give formally convergent results as each code and each method will give a slightly different turbulent realization that can at best be compared in an ensemble-averaged sense. Despite such complexities a quantitative

comparison was achieved and models generated from codes within this comparison show predictive power for actual radio VLBI observations [7]. A refinement of this study is now underway and includes other physical regimes and a deeper analysis of statistical properties.

5.3 Completion of the GRRT Code Comparison for the EHT

Another comparison of radiative transfer codes has now been accepted by the journal [9]. This includes the BHOSS code that we will use in this project as well.

The comparison, see Figs. 5 and 4 indicates good agreement among codes and confirms that in particular `BHOSS` has been a good choice for our project.

6 Conclusion

The relativistic astrophysics group in Frankfurt has a number of active projects benefiting from the 44149 BBHDISKS allocation on the HazelHen cluster.

In project BBHDISKS we have now achieved matter evolutions in spacetime metrics of two black holes in orbit around each other, see e.g. Figures 1 and 2.

First, we obtain valid and spectrally accurate (i.e. solving Einstein's field equations) initial data via `Kadath` [10].

Second, we have now linked `BHAC` [14] to the `Kadath` library [10] enabling a flexible, convenient, and efficient interface between the spacetime metric solver and `BHAC`. With this coupling in place we can now evaluate the spacetime metric to spectral accuracy on any (e.g. non-uniform) numerical grid.

Third, we have successfully initialized and evolved matter configurations within a binary spacetime. Specifically, we were able to make the necessary adjustments to handle the more complicated nature of the black hole horizons.

In summary, despite some setbacks due to the shutdown of Hazelhen or difficulties with Hawk, we have made major progress and achieved key milestones. This puts us in an excellent position to execute our project BBHDISKS as soon as Hawk becomes available.

References

1. T. Bronzwaer et al., Accepted in Astronomy & Astrophysics (2018)
2. The Event Horizon Telescope Collaboration, First M87 event horizon telescope results. I. The shadow of the supermassive black hole. Astrophys. J. **875**(1), L1 (2019)
3. The Event Horizon Telescope Collaboration, First M87 event horizon telescope results. II. Array and instrumentation. Astrophys. J. **875**(1), L2 (2019)

4. The Event Horizon Telescope Collaboration, First M87 event horizon telescope results. III. Data processing and calibration. Astrophys. J. **875**(1), L3 (2019)
5. The Event Horizon Telescope Collaboration, First M87 event horizon telescope results. IV. Imaging the central supermassive black hole. Astrophys. J. **875**(1), L4 (2019)
6. The Event Horizon Telescope Collaboration, First M87 event horizon telescope results. V. Physical origin of the asymmetric ring. Astrophys. J. **875**(1), L5 (2019)
7. The Event Horizon Telescope Collaboration. First M87 event horizon telescope results. VI. The shadow and mass of the central black hole. Astrophys. J. **875**(1), L6 (2019)
8. C. Goddi, H. Falcke, M. Kramer, L. Rezzolla et al., BlackHoleCam: fundamental physics of the galactic center. Int. J, Mod. Phys. D **26**, 1730001–239 (2017)
9. R. Gold et al., Verification of radiative transfer schemes for the EHT. Accepted by the Astrophys. J. (2020)
10. P. Grandclement, Kadath: a spectral solver for theoretical physics. J. Comput. Phys. **229**, 3334–3357 (2010)
11. R. Konoplya, L. Rezzolla, A. Zhidenko, General parametrization of axisymmetric black holes in metric theories of gravity. Phys. Rev. D **93**(6), 064015 (2016)
12. Y. Mizuno et al., The current ability to test theories of gravity with black hole shadows. Nat. Astron. **2**, 585–590 (2018)
13. H.P. Pfeiffer, J.W. York, Extrinsic curvature and the Einstein constraints. Phys. Rev. D **67**, 044022 (2003)
14. O. Porth, et al., The Event Horizon Telescope Collaboration, The event horizon general relativistic magnetohydrodynamic code comparison project. arXiv e-prints, April 2019
15. O. Porth et al., The black hole accretion code. Comput. Astrophys. Cosmol. **4**, 1 (2017)
16. O. Porth, C. Xia, T. Hendrix, S.P. Moschou, R. Keppens, MPI-AMRVAC for solar and astrophysics. Astrophys. J. Supp. **214**, 4 (2014)
17. L. Rezzolla, A. Zhidenko, New parametrization for spherically symmetric black holes in metric theories of gravity. Phys. Rev. D **90**(8), 084009 (2014)
18. Z. Younsi, et al. (2019, in preparation)

Molecules, Interfaces, and Solids

Holger Fehske and Christoph van Wüllen

The following chapter reveals that science in the field of molecules, interfaces and solids has profited substantially from the computational resources provided by the High Performance Computing Center Stuttgart and the Steinbuch Centre for Computing Karlsruhe. As only one-third of the very interesting and promising contributions could be captured in this part, we have selected rather diverse activities, from both a scientific and methodological point of view, not least to demonstrate that a great variety of techniques has been used to reach scientific progress in various areas of chemistry, physics and material science research.

We start with the contribution of the Meyer group in Heidelberg, which is interested in the Eigen cation $(H_3O) + (H_2O)_3$. While this may appear as a "small system" to many readers, its 33 degrees of freedom are a formidable challenge to a full quantum mechanical treatment of its nuclear motion. The MCTDH (multi configuration time-dependent Hartree) program developed in Heidelberg allows for an efficient quantum mechanical treatment, but this can only be fully exploited if the high-dimensional potential energy of the molecule (a function of 33 variables) is transformed into a canonical tensor form. This has been achieved with a new program written in this work, a program that made use of up to 19200 CPU cores. With this, a detailed and complete quantum mechanical treatment of nuclear motion in the Eigen cation has been performed for the first time, and this small system is a test-tube for the dynamics of proton transfers in aqueous solutions.

H. Fehske
Institut für Physik, Lehrstuhl Komplexe Quantensysteme,
Universität Greifswald, Felix-Hausdorff-Str. 6, 17489 Greifswald, Germany
e-mail: fehske@physik.uni-greifswald.de

C. van Wüllen
Fachbereich Chemie, Technische Universität Kaiserslautern
Erwin-Schrödinger-Str. 52, 67663 Kaiserslautern, Germany
e-mail: vanwullen@chemie.uni-kl.de

The contribution from Lesnicki and Sulpizi from the university of Mainz also targets at dynamics in liquid water, but here we have a much larger system size and consequently a much less detailed simulation method, a molecular dynamics (MD) simulation in this case. Water at mineral interfaces behaves quite differently from bulk water, mostly because the ionic interface leads to a preferred orientation within the first few layers of water molecules and therefore makes the hydrogen-bonding network stiffer. To investigate vibrational relaxation at the mineral-water interface, the group used a new simulation protocol: kinetic energy is put into the atoms close to the interface (this is easily done in the computer!) and it is monitored how this excess kinetic energy "flows" into the bulk. While this is not what happens in the 2D sum frequency generation experiment, the primary output, namely the time constant for vibration energy relaxation, is the same. Interesting details emerge from the analysis of the MD data, for example different timescales in different (vibrational) frequency ranges, and these findings can directly be compared to experiment.

A joint contribution from the universities of Marburg and Stuttgart investigates confinement effects in molecular heterogeneous catalysis, using large-scale MD simulations. The force field methods used here cannot model the catalytic transformation, since this involves bond breaking and making. But everything happens in a confined space (e.g. a mesopore) where the catalyst is immobilized, and the transport of the substrate to and of the product from the catalyst is of prime importance, and this is the main target of the MD simulations performed here. The ring-closing metathesis of a large ω, ω' diolefin by a ruthenium-based (Grubbs-Hoveyda) catalyst was investigated, and both dynamic (diffusion constants etc.) as well as static (enrichment/depletion of substrate and product in different portions of the pore) properties have been extracted from the calculations. For example, it has been found that the product enriches near the catalytic centers which is detrimental to catalytic efficiency. These calculations therefore offer a handle for a rational design of mesoporous systems with immobilized catalysts.

The physics projects introduced in the following will mainly address the striking dynamical and transport properties of electronically low-dimensional systems: zero-dimensional quantum dots, one-dimensional nano-wires and two-dimensional (film) superconductors.

On outstanding example in this respect is the work by Philipp W. Schering and Götz S. Uhrig from the Solid State Theory Group at Dortmund University who discussed the spin dynamics of an (inhomogeneous) ensemble of GaAs quantum dots subjected to trains of periodic optical pulses. Their study paves the way to a better understanding of very recent pump-probe experiments, e.g., with respect to the fascinating phenomenon of nuclei-induced frequency focusing, and even allows to predict new effects, such as the emergence of resonant spin amplification in a Faraday geometry. The latter gives access to the longitudinal g factor of the charge carriers localized in the quantum dots. Another interesting result is that an increase of the pumping strength leads to a decrease of the spin relaxation time in the system. A prerequisite for all these findings are the large-scale simulations

performed on Hazel Hen, making it possible to reach the non-equilibrium states for the full range of magnetic fields used in experiments. Thereby the authors resort on a well-established semiclassical approach to the problem, supplemented by a more realistic description of the nuclear spin bath. Technically they performed an adaptive integration of the system of ordinary differential equations by the Dormand-Prince method, which is easily MPI-parallelizable with almost ideal speedup.

Giessen's University and Center for Materials Research group headed by S. Sanna explored the subtle interplay between atomic structure and electronic states in the process of wire-growth on high-index Si(*hkl*)-Au (stepped) surfaces. The investigations focus on the impact of H adsorption on the electronic structure with the objective to tune the system's metallicity. To come to the point: The study reveals that the influence of atomic hydrogen goes far beyond simple charge transfer. Thus it could be shown that H adsorption on the energetically most favorable sites at the Si step edge not only leads to modifications in the electronic structure due to adsorbate-to-substrate charge transfer, resulting, e.g., in a band shift and gap widening, but also changes the character of the hybridization by affecting the spatial distribution of the wave functions on the unoccupied states. As a result the plasmonic excitations will be strongly influenced by the unoccupied band structure. The presented first-principle calculations of the electronic band structures, potential energy surfaces and squared wave functions are based on total energy density functional calculations within the generalized gradient approximation, using the VASP simulation package, where best performance was reached with pure MPI parallelization.

Motivated by scanning tunneling microscope measurements conducted by the Wulfhekel KIT group on the surface of superconducting Al(111) in the presence of an adatom, F. Evers and M. Stosiek from the University of Regensburg investigated thin film superconductors with a single impurity. Here the central issue is the predicted enhancement of superconductivity (critical temperature, mean superconducting gap) by disorder, which could be verified by the authors analyzing the statistical properties of the local density of states and the local gap function, as well as the spatial distribution of the pairing amplitude within their self-consistent numerical simulation scheme. Equally interesting the authors demonstrated that the formation of superconducting islands is an indispensable ingredient in any theoretical approach aiming to describe the observed superconductor-insulator transition. This effect, however, was ignored in the analytical studies so far. The many-body localization (MBL) transition is another ambitious, disorder-related problem addressed in this project. With the help of the ForHLR2 computing resources, Evers and Stosiek have analyzed dynamical signatures of this transition for a one-dimensional quantum lattice model with random potentials contributions. Combining very efficient Chebyshev expansion and Kernel polynomial techniques with an optimized sparse matrix-vector-multiplication implementation finite-size effects could be overcome. This allows to discuss the level-spacing distribution more seriously and so a further (intermediate) MBL phase in addition to the actual MBL state has been detected.

In summary, almost all projects supported during the period under review, have in common, besides a high scientific quality, the strong need for computers with high performance to achieve their results. Therefore, the supercomputing facilities at HLRS and KIT-SCC have been essential for their success.

Calculation of Global, High-Dimensional Potential Energy Surface Fits in Canonical Decomposition Form Using Monte-Carlo Methods: Application to the Eigen Cation

Markus Schröder, Hans-Dieter Meyer, and Oriol Vendrell

Abstract We have implemented a Monte-Carlo version of a well-known alternating least squares algorithm to obtain a sum-of-products representation of potential energy surfaces, more precisely a so-called Canonical Polyadic Decomposition, for use in quantum-dynamical simulations. Our modification replaces exact integrals with Monte-Carlo integrals. The incorporation of correlated weights, and hence weighted integrals, is straight forward using importance sampling. Using Monte-Carlo methods allows to efficiently solve high-dimensional integrals that are needed in the original scheme and enables us to treat much larger systems than previously possible. We demonstrate the method with calculations on the 33-dimensional Eigen cation.

1 Introduction

In the recent years there has been rapid developments in the field of molecular quantum dynamics. In particular, many efforts have been made to treat ever larger molecules with many degrees of freedom on a quantum mechanical level. To study such a system one usually needs to solve the time-dependent Schrödinger equation

$$\frac{\partial}{\partial t}\Psi(q_1, ..., q_f, t) = -i\hat{H}\Psi(q_1, ..., q_f, t) \tag{1}$$

($\hbar = 1$) where $\hat{H} = \hat{T} + V$ is the system Hamiltonian with the kinetic energy $\hat{T}$ and a potential V. The state of the system is described by the wavefunction Ψ which

M. Schröder · H.-D. Meyer (✉) · O. Vendrell
Physikalisch-Chemisches Institut, Universität Heidelberg, Im Neuenheimer Feld 229, 69120 Heidelberg, Germany
e-mail: hans-dieter.meyer@pci.uni-heidelberg.de

M. Schröder
e-mail: markus.schroeder@pci.uni-heidelberg.de

O. Vendrell
e-mail: oriol.vendrell@pci.uni-heidelberg.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_5

depends on the physical coordinate q_κ, where $\kappa = 1 \cdots f$ labels the f degrees of freedom (DOF) of the system.

When the Schrödinger equation is to be solved for a system with many atoms, as is the case in molecular systems, the main obstacle to overcome is the so-called "curse of dimensionality": usually the wavefunction is sampled on a product grid of primitive basis functions, usually just grid points in coordinate space. One selects a set of, say N, grid points for each of the f (physical) coordinates or DOF of the system and then samples the wavefunction on the N^f-dimensional product space of the single coordinates. This implies that N^f coefficients must be stored to represent the wavefunction. The coefficients which represent the wavefunction in coordinate space can then be interpreted as an $f-$way tensor.

One easily sees that this ansatz will quickly lead to an exhaustion of numerical resources even for small systems because of the exponential scaling law of the required memory. Within the Heidelberg group, we therefore have developed the multi-configuration time-dependent Hartree (MCTDH) algorithm [1–6] to compact the amount of information that needs to be stored and processed while at the same time maintain a closed set of equations of motion of the systems wavefunction. The algorithm is based on a Tucker decomposition of the wavefunction tensor. To be specific, the wavefunction is approximated as

$$\Psi(q_1, ..., q_f, t) \approx \sum_{j_1}^{n_1} \cdots \sum_{j_f}^{n_f} A_{j_1 \cdots j_f}(t) \cdot \phi_{j_1}^{(1)}(q_1, t) \cdots \phi_{j_f}^{(f)}(q_f, t) \tag{2}$$

or, when sampled on the primitive coordinate grids,

$$\Psi_{i_1, \cdots, i_f}(t) \approx \sum_{j_1}^{n_1} \cdots \sum_{j_f}^{n_f} A_{j_1 \cdots j_f}(t) \phi_{j_1, i_i}^{(1)}(t) \cdots \phi_{j_f, i_f}^{(f)}(t). \tag{3}$$

Here, $A_{j_1 \cdots j_f}$ is the Tucker core tensor, $\phi^{(\kappa)}$ are time-dependent basis functions and n_κ are the expansion orders in the coordinates q_κ. This ansatz shifts the exponential scaling law to the core tensor, however, now with a smaller base n_κ.

One can further compact the core tensor by combining a number of physical coordinates into logical coordinates Q_κ. This is equivalent to combining a few indices of the $f-$way tensor into larger ones, thereby turning the tensor into a $p-$way tensor with $p \leq f$ and then performing the Tucker decomposition:

$$\Psi(q_1, ..., q_f, t) = \Psi(Q_1, ..., Q_p, t) \approx \sum_{j_1}^{n_1} \cdots \sum_{j_p}^{n_p} A_{j_1 \cdots j_p}(t) \cdot \phi_{j_1}^{(1)}(Q_1, t) \cdots \phi_{j_p}^{(p)}(Q_f, t), \tag{4}$$

where p is the number of modes or "logical coordinates". The mode-combination scheme now opens the path to a hierarchical tensor decomposition as one may see the now multi-dimensional basis functions $\phi^{(\kappa)}$ again as tensors which can be decom-

posed into Tucker Format. This gives raise to the multi-layer multi-configuration time-dependent Hartree (ML-MCTDH) approach [6–11]) for which also a hierarchical set of coupled equations of motion can be derived which optimally resemble the time-dependent Schrödinger equation Eq. (1).

Both, MCTDH and ML-MCTDH have in common that the equations of motions are coupled through so-called "mean fields". The mean-fields are contractions of the wavefunction tensor and the system Hamiltonian over all but one (logical) indices. Hence, ideally, the system Hamiltonian should also be represented in a compatible format to the wavefunction tensor. Compatible here means that the Hamiltonian should be in a sum-of-products form

$$\hat{H} = \sum_{r=1}^{s} c_r \prod_{\kappa=1}^{p} \hat{h}_r^{(\kappa)}(Q_\kappa), \tag{5}$$

where the single-particle Hamiltonians $\hat{h}_r^{(\kappa)}$ now exclusively depend on the coordinate $Q\kappa$. This is usually automatically fulfilled for the kinetic energy, but not for the potential energy. The potential is usually provided to us as a numerical routine which takes a coordinate vector as argument and returns an energy value. In this form it cannot be used directly within the Heidelberg implementation of MCTDH. Realistic potentials must therefore undergo a pre-processing step in which they are transformed into a numerical sum-of-products representation. More precisely, the potential needs to be transformed and stored in a tensor format prior to carrying out the dynamics calculations.

One of best known algorithms to perform this transformation is the Potfit algorithm [3, 12, 13] and its variants [14–17]. Within Potfit, the potential V is decomposed into a Tucker format, very similar to the wavefunction:

$$V(q_1 \dots q_f) = \sum_{j_1=1}^{m_1} \cdots \sum_{j_p=1}^{m_p} C_{j_1,\dots,j_p}\, v_{j_1}^{(1)}(Q_1) \cdots v_{j_p}^{(p)}(Q_p)\,. \tag{6}$$

where $C_{j_1,\dots,j_p}$ is again the Tucker core tensor and $v_{j_\kappa}^{(\kappa)}(Q_\kappa)$ are orthogonal basis functions. Obtaining this decomposition, however, is quite labor intensive if there are more than 6 DOFs, as it requires multiple contractions over the complete primitive grid. Recent algorithms therefore resorted to incomplete contractions via random sampling [16, 17] or sparse grid techniques [14]. Random sampling will also be used in the current contribution.

Apart from the Tucker format, however, there also exists the so-called Canonical Polyadic Decomposition (CPD) format, sometimes called "CANDECOMP" or "PARAFAC" in the literature [18]. The CPD format promises a higher compression rate than the Tucker format, however, is also much harder to obtain. Being able to obtain a CPD form of the potential would therefore open the path to study very large systems as will be demonstrated below. The main focus of this contribution hence lies

in obtaining a decomposition of the potential in CPD form. The algorithm outlined below has been recently published in Ref. [19].

2 Canonical Polyadic Decomposition

Using mode-combinations the potential in CPD form is given as

$$V(Q_1 \cdots Q_p) \approx \sum_{r=1}^{R} c_r \nu_r^{(1)}(Q_1) \cdots \nu_r^{(p)}(Q_p), \tag{7}$$

where c_r are expansion coefficients and $\nu_r^{(\kappa)}$ are expansion functions. Unlike the Tucker format, there are in general no restrictions on the expansion functions. In particular, it is not demanded that the expansion functions form an orthogonal set. Often, and also in this work, it is, however, demanded that the expansion functions are normalized, which gives raise to expansion coefficients as in Eq. (7).

2.1 Alternating Least Squares

Given the expression, Eq. (7), for the decomposition of the Potential, the task is then to find optimal expansion functions and coefficients such that the error with respect to the exact potential is minimized. Due to the relaxed restrictions on the expansion functions this is a highly nonlinear task.

In the literature, often the so-called alternating least squares method is used which is outlined in a very basic form below. One starts with a functional that is to be minimized:

$$J = \sum_I \left(V_I - V_I^{\mathrm{CPD}}\right)^2 + \epsilon \sum_r c_r^2 \sum_I \Omega_{r,I}^2. \tag{8}$$

Here I is the composite index $\{i_1, \ldots, i_p\}$ and $\Omega_{r,I} = \prod_\kappa \nu_{r,i_\kappa}^{(\kappa)}$ with $\nu_{r,i_\kappa}^{(\kappa)}$ being the rth basis function for the logical coordinate Q_κ evaluated at the i_κ'th grid point of Q_κ. The first part of Eq. (8) measures the error of the expansion with respect to the exact potential while the second part is called the regularization with regularization parameter ϵ as will become apparent below.

To arrive at expressions for obtaining the basis functions and coefficients, one calculates the functional derivative of J as

$$\frac{\delta J}{\delta c_r \nu_r^{(\kappa)}} = -2 \sum_{I^\kappa} V_I \Omega_{r,I^\kappa}^\kappa + 2 \sum_{r'} c_{r'} \nu_{r'}^{(\kappa)} S_{r,r'}^{(\kappa)} + 2\epsilon c_r \nu_r^{(\kappa)} S_{r,r}^{(\kappa)} \longrightarrow 0. \tag{9}$$

where the positive semi definite matrix

$$S^{(\kappa)}_{r,r'} = \sum_{I^\kappa} \Omega^\kappa_{r,I^\kappa} \Omega^\kappa_{r',I^\kappa}. \tag{10}$$

denotes, with $\Omega^\kappa_{r,I} = \prod_{\kappa' \neq \kappa} \nu^{(\kappa')}_{r,i'_\kappa}$, the overlap of the product of all basis functions but the κth one. Note, that the sum is now done over the index I^κ which is the composite index of all indices but the κth one. One can now define

$$\begin{aligned} x^{(\kappa)}_{r,i_\kappa} &= c_r \nu^{(\kappa)}_{r,i_\kappa} \\ b^{(\kappa)}_{r,i_\kappa} &= \sum_{I^\kappa} V_I \Omega^\kappa_{r,I^\kappa} \end{aligned} \tag{11}$$

and arrive at the working equations for obtaining optimal coefficients and basis functions for one logical coordinate:

$$b^{(\kappa)}_{r,i_\kappa} = \sum_{r'} \left(S^{(\kappa)}_{r,r'} + \epsilon \delta_{r,r'} \right) x^{(\kappa)}_{r',i_\kappa}, \tag{12}$$

where $S^{(\kappa)}_{r,r} = 1$ has been used for the regularization part. After choosing the regularization parameter, Eq. (12) can be solved with standard linear algebra tools.

Since the solution of Eq. (12) for one mode, however, depends on the solution of the same equation for all other modes, one resorts to an iterative scheme: One starts with an initial guess for the basis functions and coefficients, usually just random numbers, and iterates until some stop criterion is met. This procedure is called the Alternating Least Squares (ALS) method.

2.2 Monte-Carlo – ALS

In the simple form outlined above, the ALS algorithm already produces very usable results. It exhibits, however, one serious drawback: The quantities $S^{(\kappa)}$ and $b^{(\kappa)}$ both are formed through contractions over all but one degrees of freedom. While the contraction is separable for $S^{(\kappa)}$, this is not the case for $b^{(\kappa)}$ such that here the sum must really (and repeatedly) visit all combinations of primitive grid points. This seriously limits the applicability of the ALS algorithm to approximately 8–10 DOF at most. To mitigate this drawback one can resort to a sparse sampling of the contractions. In the following this will be achieved thought a Monte-Carlo sampling.

To this end, one needs to adjust the functional, Eq. (8) to account for the sampling. Moreover, a slightly different functional is needed for every DOF because only the modes over which the contraction is carried out need to participate in the sampling. Hence, one introduces a functional for each mode κ as

$$J_\kappa^{\tilde{\mathrm{W}}} = \sum_I \tilde{W}_I^{(\kappa)} \left(V_I - V_I^{\mathrm{CPD}}\right)^2 + \epsilon \sum_r c_r^2 \sum_I \tilde{W}_I^{(\kappa)} \Omega_{r,I}^2. \tag{13}$$

The difference to Eq. (8) lies in the presence of a weight function $\tilde{W}_I^{(\kappa)}$ with

$$\tilde{W}_I^{(\kappa)} = w_{i_\kappa}^{(\kappa)} W_{I^\kappa}^\kappa, \tag{14}$$

i.e., the weight function $\tilde{W}^{(\kappa)}$ is approximated by a weight function that depends only on the coordinate κ times a weight function that depends on all but the coordinate κ.

Performing the functional derivative as before then leads to

$$\begin{aligned}\frac{\delta J_\kappa^{\mathrm{W}}}{\delta c_r v_r^{(\kappa)}} &= -2w^{(\kappa)} \sum_{I^\kappa} W_{I^\kappa}^\kappa V_I \Omega_{r,I^\kappa}^\kappa + 2w^{(\kappa)} \sum_{r'} c_{r'} v_{r'}^{(\kappa)} Z_{r,r'}^{(\kappa)} \\ &\quad + 2\epsilon w^{(\kappa)} c_r v_r^{(\kappa)} Z_{r,r}^{(\kappa)} \longrightarrow 0,\end{aligned} \tag{15}$$

with

$$Z_{r,r'}^{(\kappa)} = \sum_{I^\kappa} \Omega_{r,I^\kappa}^\kappa W_{I^\kappa}^\kappa \Omega_{r',I^\kappa}^\kappa. \tag{16}$$

As in the previous subsection, Eq. (15) now leads to simplified expressions:

$$d_{r,i_\kappa}^{(\kappa)} = \sum_{r'} Z_{r,r'}^{(\kappa)} \left(1 + \epsilon \delta_{r,r'}\right) x_{r',i_\kappa}^{(\kappa)}, \tag{17}$$

where the quantities

$$\begin{aligned} x_{r,i_\kappa}^{(\kappa)} &= c_r v_{r,i_\kappa}^{(\kappa)} \\ d_{r,i_\kappa}^{(\kappa)} &= \sum_{I^\kappa} W_{I^\kappa}^\kappa V_I \Omega_{r,I^\kappa}^\kappa \end{aligned} \tag{18}$$

have been used. Equation (17) again can be solved with standard linear algebra tools. Note, that the weight function $w^{(\kappa)}$ drops out of the working equations and one is left with the same contractions as before, but now with respect to a weight function W^κ. Note, that in this form the contraction still requires visiting all primitive basis points. Moreover, the presence of the weight function spoils the separability of the overlap matrix.

In the following, therefore, the weight function will be interpreted as a distribution function of sampling points and the contraction is approximated by Monte-Carlo sampling [20]:

$$\sum_I W_I F_I \approx \mathcal{N} \sum_s^{N_C} F_s, \tag{19}$$

where s labels the Monte-Carlo points and N_C is the number of sampling points.

Applying Eq. (19) to Eq. (17) one arrives at the approximate expressions

$$
\begin{aligned}
Z^{(\kappa)}_{r,r'} &\approx \sum_{s^\kappa} \Omega^{\kappa}_{r,s^\kappa} \Omega^{\kappa}_{r',s^\kappa}, \\
d^{(\kappa)}_{r,i_\kappa} &\approx \sum_{s^\kappa} V_{i_\kappa,s^\kappa} \Omega^{\kappa}_{r,s^\kappa}.
\end{aligned}
\tag{20}
$$

Here, the index s^κ denotes a Monte-Carlo index that denotes sampling points for all DOF except the κth one. Note that the quantities $d^{(\kappa)}$ are constructed from 1-D cuts (one index is complete) through the V-tensor and that the matrix $Z^{(\kappa)}$ is symmetric. The matrix elements of $Z^{(\kappa)}$ are technically large dot-products.

3 Implementation Details

Equations (20) are in general the working equations that have been implemented in a new program "mccpd" within the Heidelberg MCTDH package. The program hast been used to obtain the CPD fit of the potential for the calculations on the Eigen cation as outlined below. The size of the problem, and also the large numerical demand of the numerical routine that was used to evaluate the potential energy made it imperative to organize the program such that many quantities can be re-used and the calculations are optimized.

There are two major blocks of work that need to be performed for obtaining a CPD tensor with the method outlined above. The first block is to evaluate the potential along the 1-D cuts through the sampling points as needed in the second line of Eq. (20). This part has to be done only once and serves to collect data about the potential. There are cases where molecular symmetries can be exploited, which is outlined in detail in Ref. [19]. In this case symmetric CPD expansions can be created. Symmetrically equivalent configurations and energies (and hence 1-D cuts) can be constructed via symmetry operations (in practice just permutations of primitive basis points) instead of calling the potential routine again. This could, however, only be used to a very limited degree in the present case because symmetry adapted internal coordinates could not be used.

The second major part of the work is performing the ALS iterations. One may recognize that during the iterations all quantities, the $Z^{(\kappa)}$- and $d^{(\kappa)}$-matrices need to be build anew for each mode. Here, a number of optimizations are possible as outlined below.

Finally, after the ALS iterations are complete, the fit is tested on an independent and usually much larger set of sampling points for accuracy. This requires calling the potential routine once for each sampling point and comparing it to the value on the tensor. This task is also performed in parallel by distributing the sampling points among the MPI tasks. For the calculations inside the MPI rank again shared memory parallelization can be used.

3.1 Parallelization

The program supports both, shared memory parallelization with OpenMP and distributed memory parallelization with MPI. The latter is mainly used to distribute the sampling points among the MPI ranks such that each rank possesses a local set of sampling points. Each MPI rank, therefore, must also possess the matching set of 1-D cuts. Both can either be read-in or created as needed. If the potential energy routine is thread-safe, it can be called using shared memory parallelization for generating the cuts. Unfortunately this was not the case for the potential routine used in the present case such that pure distributed memory parallelization has been used.

The 1-D-cuts are stored as column major (Fortran order) such that, if shared memory parallelization is possible, each thread calculates a chunk of columns where the chuck size can be controlled with environment variables. The cuts and sampling points are not communicated except for reading/writing on persistent memory, if required.

Within the second major part, the ALS iterations, each MPI rank possesses a fixed subset of sampling points (which is usually the same as for the first part). Shared memory parallelization is here used to parallelize generating the $Z^{(\kappa)}$- and $d^{(\kappa)}$-matrices from the local subset of sampling points. These quantities must then be reduced among the MPI ranks. Also, shared memory parallelization, and optionally distributed memory parallelization using Scalapack, can be used to solve the linear system, Eq. (17) by linking to appropriate and optimized numerical libraries.

3.2 Pre-fetching and Re-using Sampling Points

The distribution function of the sampling points as given in Eq. (14) can be created from a single weight function, or a single set of sampling points, according to

$$W^{\kappa}_{I_\kappa} = \sum_{i_\kappa} W_I. \tag{21}$$

As a consequence, all sets of sampling points for the various modes only differ in the index that is left out. The sampling points of the common modes are the same. This allows pre-fetching and storing the values of the basis functions at the sampling points. Since one basis function $\nu^{(\kappa)}_r$ usually fits into the CPU cache, picking sampling points for a particular combination of r and κ is very efficient. These points can be stored in a linearized and continuous sampling cache. Moreover, these sampling-caches only need to be updated after an ALS sweep over the respective coordinate but can be used for every mode during an ALS sweep. From the continuous sampling caches the Ω-matrix, and from its transposed the $Z^{(\kappa)}$- and $d^{(\kappa)}$-matrices can then be build very efficiently.

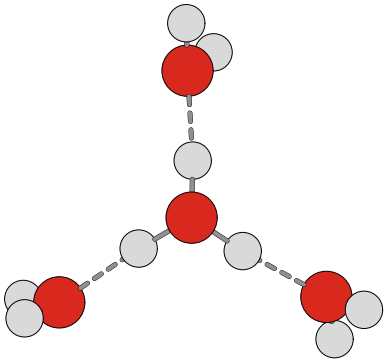


Fig. 1 The Eigen cation: a central, positively charged hydronium is solvated in a shell consisting of three water molecules

3.3 Growing the CPD Tensor

Unfortunately, one of the major drawbacks of the ALS algorithm is that it converges rather slowly. More precisely, it can be shown that the original algorithm improves monotonically, but may never converge towards a limit [21]. The Monte-Carlo version of the ALS algorithm may only converge for an initial number of iterations until non-monotonic behavior sets in. It seems to be very helpful, therefore, to start with a rather small tensor, typically a few hundred terms, iterate the optimization routine a number of times and subsequently grow the tensor by adding additional terms which are initialized with random numbers but zero coefficient. This usually accelerates the reduction of the fit error quite substantially. This strategy can be used repeatedly until either a desired accuracy is achieved or a maximum number of terms has been reached.

4 Application to the Eigen Cation

The protonated water tetramer $H_9O_4^+$, or Eigen cation, as depicted in Fig. 1, is besides the Zundel cation one of the most important protonated water clusters. $H_9O_4^+$ is an extremely floppy molecule that is very hard to describe. Recently, experimental absorption spectra [22, 23] have been obtained for the Eigen cation in the range of $200\,\mathrm{cm}^{-1}$ and $4000\,\mathrm{cm}^{-1}$. While some of the features of the spectrum have been identified before, there is still an open debate about the origin of the broadening of the peak at $2700\,\mathrm{cm}^{-1}$. It is clear, that it is associated with the transfer of the protons between the central oxygen and one of the water molecules. However, other states must be involved as well.

To our knowledge, the quantum calculation of the infrared spectrum in the complete spectral range of one-photon absorptions between 0 to $4000\,\mathrm{cm}^{-1}$ and with full inclusion of anharmonicity and coupling, has not yet been achieved for this very challenging system. The difficulty lies in the high dimensionality of the system: the

molecule consists of $N = 13$ atoms, and hence $3N - 6 = 33$ internal DOF. While the treatment of the systems wavefunction with help of the ML-MCTDH algorithms has been in principle possible for a number of years, there has (to our knowledge) so far been no tool to transform high-dimensional potential energy surfaces into a sum-of-products form. With the help of the aforementioned techniques, in particular with the global expansion of the potential according to Eq. (7), this goal could be achieved such that an accurate modeling of the systems absorption spectrum was possible.

The absorption spectrum within the linear response regime is given by the Fourier transform of the dipole-dipole- correlation function $C_\mu(t)$ as

$$\alpha(\omega) \propto \omega \int_{-\infty}^{\infty} e^{i\omega t} C_{\mu\mu}(t) \tag{22}$$

with

$$C_{\mu\mu}(t) = \frac{1}{3} \sum_{i=x,y,z} \langle \psi_0 | \, \mu_i e^{-i(\hat{H}-E_0)t} \mu_i \, | \psi_0 \rangle \tag{23}$$

and ψ_0 being the ground state with energy E_0, and μ_x, μ_y, and μ_z are the dipole moment surfaces in the x-, y-, and z-direction in the laboratory system. Note, that the μ_i are, like the potential, 33-dimensional surfaces as they depend on all 33 internal coordinates. The averaging over the dipole components in Eq. (23) is performed to model the random orientations of the molecules in the spectroscopic experiments.

4.1 Preparation of the Dynamical Calculations

Equation (23) now implies that in order to obtain the spectrum, one needs to solve the time-dependent Schrödinger equation for the initial states $\left|\psi_{\mu_i}\right\rangle = \mu_i \, |\psi_0\rangle$. To set up a numerical Hamiltonian for solving the time-dependent Schrödinger equation one first needs to chose a set of suitable internal coordinates. For the Eigen cation we used so-called polyspherical coordinates which have the advantage that the kinetic energy operator $\hat{T}$ is automatically given in a sum-of-products form, Eq. (5), as required by the MCTDH implementation. Once having defined the internal polyspherical coordinates, analytic expressions for kinetic energy can be calculated with the TANA program package by D. Lauvergnat [24].

In a second step we combined the $f = 33$ physical DOF into $p = 13$ logical ones where the combination was performed based on geometrical and physical considerations, see Table 1.

Table 1 Mode combinations, grid size, and physical meaning of the internal coordinates of the Eigen cation. The labels A, B, C refer each to one of the three "arms" of the molecule. In the last line the size of the total primitive grid is given.

Mode	Physical coord.	No. points	Remark
1	u_{AB}, u_{AC}, α	$11 \times 11 \times 11 = 1331$	Global O-O angles and pyramidalization
2	R_A, z_A	$13 \times 9 = 117$	Distance Water A and proton A from center
3	R_B, z_B	$13 \times 9 = 117$	Distance Water B and proton B from center
4	R_C, z_C	$13 \times 9 = 117$	Distance Water C and proton C from center
5	R_{1A}, R_{2A}, θ_A	$9 \times 7 \times 7 = 441$	internal Water A
6	R_{1B}, R_{2B}, θ_B	$9 \times 7 \times 7 = 441$	Internal Water B
7	R_{1C}, R_{2C}, θ_C	$9 \times 7 \times 7 = 441$	Internal Water C
8	γ_A, u_A	$13 \times 9 = 117$	Wagging and rocking of water A
9	γ_B, u_B	$13 \times 9 = 117$	Wagging and rocking of water B
10	γ_C, u_C	$13 \times 9 = 117$	Wagging and rocking of water C
11	x_A, y_A, α_A	$9 \times 9 \times 11 = 891$	Proton A in-/out-of-plane and water A rotation
12	x_B, y_B, α_B	$9 \times 9 \times 11 = 891$	Proton B in-/out-of-plane and water B rotation
13	x_C, y_C, α_C	$9 \times 9 \times 11 = 891$	Proton C in-/out-of-plane and water C rotation
		2.1×10^{32}	Primitive grid size

In a third step one then needs to fit the potential energy V as well as the three dipole surfaces μ_i in the form of Eq. (7). To this end we first created a set of sampling points using a Metropolis-Hastings algorithm according to the weight function

$$W(Q_1, \cdots, Q_p) = \sum_i a_i \exp\left(-\beta_i V(Q_1, \cdots, Q_p)\right) \tag{24}$$

where the $\beta_i = 1/k_B T_i$ can be interpreted as inverse temperatures with k_B being the Boltzmann constant and a_i is a coefficient that weights the different temperature contributions. Altogether, 7×10^6 sampling points have been created. 2×10^6 points for each $k_B T = 500\,\mathrm{cm}^{-1}$, $k_B T = 1000\,\mathrm{cm}^{-1}$ and $k_B T = 2000\,\mathrm{cm}^{-1}$ as well as 7.5×10^5 sampling points for $k_B T = 3000\,\mathrm{cm}^{-1}$ and 2.5×10^5 sampling points for $k_B T = 4000\,\mathrm{cm}^{-1}$. These points have been calculated as 9600 independent trajectories (of which only one in 100 points was accepted as a valid sampling points to reduce the correlation inside the trajectories) in parallel, one trajectory on one CPU core with a wall-time of approximately one hour. The same set of sampling points was used to fit all above mentioned quantities.

After the sampling is obtained, one of the major tasks is to calculate the 1-D potential cuts as needed in Eq. (20). Here, a total number of 6.1×10^{10} calls to the potential energy routine have been preformed. Note, that this is almost 22 orders of magnitude less than the total number of primitive grid points, cf. Table 1. This task was performed using between 9600 and 19200 CPU cores in parallel, typically with a wall-time below four hours. As all cuts can be calculated independently a re-start of the calculation is easily possible at any time. For all three quantities, V and μ_x,

μ_y, μ_z, the ALS optimization using Eq. (17) was done in one chunk on 9600 cores for the potential (wall-time: 22h) and 4800 cores for the μ_i (wall-time: 4 h). The reason for the different amount of work is the number of terms (cf. Eq. (7)) in the expansions where $R = 2048$ was used for the potential and $R = 1024$ for the μ_i. The amount of work to construct the matrix Z in Eq. (17) is here dominating the numerical efforts and scales as R^2. For this reason, also fewer iterations have been used to create the potential fit.

The fits where subsequently tested on an independent set of sampling points generated as described before. It was found that the root-mean-square error for temperatures $k_\mathrm{B}T \leq 2000\ \mathrm{cm}^{-1}$ was below $150\,\mathrm{cm}^{-1}$ which is an extremely promising value for this large a system. For the dipole surfaces, which have been fitted with less accuracy because only spectral peak heights and not positions are influenced by errors, the error obtained was below 2% for temperatures $k_\mathrm{B}T \leq 2000\,\mathrm{cm}^{-1}$.

4.2 Numerical Results

Once all constituents to obtain the spectrum, Eq. (22) have been obtained, the correlation function was calculated by first operating the dipoles on the ground state of the Eigen cation and subsequently propagating the resulting states in time. These calculations were performed on workstations outside the HLRS as solving the time-dependent Schrödinger equation in a hierarchical Tucker format is highly involved and distributed memory parallelization has not yet been implemented such that long wall-times are needed. The resulting calculated spectrum is depicted in the lower panel of Fig. 2 together with experimental spectra in the upper panel. The calculated spectrum was shifted $60\,\mathrm{cm}^{-1}$ towards lower energies to match the positions of the two prominent peaks around $250\,\mathrm{cm}^{-1}$ with the experimentally obtained ones.

One can see that all major features of the experimental spectrum are present also in the calculated ones. In particular, the main features at $250\,\mathrm{cm}^{-1}$ which have been identified previously as wagging modes and O-O distance mode of the marginal water molecules are clearly obtained, along with the broad feature between 2500 and $3000\,\mathrm{cm}^{-1}$. This peak is associated with the motion of the three inner protons along their respective O-O direction. Another prominent feature in the experimental spectrum are the symmetric and antisymmetric stretching modes of the marginal water molecules between 3500 and $4000\,\mathrm{cm}^{-1}$. These were obtained at somewhat higher energies and with different intensities as compared to the experiment.

Note further, that besides the main features a number of detailed features of the experimental spectra could also be obtained, as for instance the rather broad peak at $1000\,\mathrm{cm}^{-1}$ and two of the three peaks between 1500 and $2000\,\mathrm{cm}^{-1}$. The triple peak at $2250\,\mathrm{cm}^{-1}$ in the experimental spectrum seams also to be present in the calculated spectrum, however, not completely resolved and shifted towards slightly higher energies.

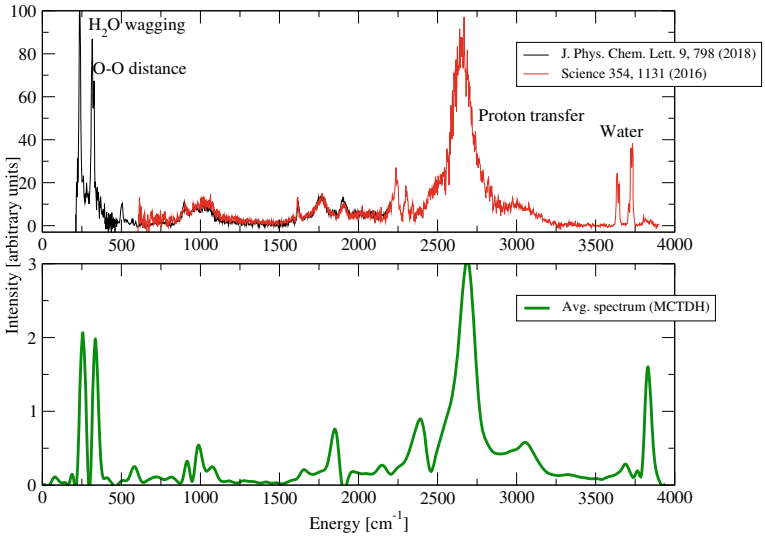


Fig. 2 Spectrum of the Eigen cation. Upper panel: experimental data from Refs. [22] (red line) and [23] (black line). Lower panel: calculated spectrum with MCTDH (shifted by 60 cm^{-1} towards lower energies to match peak positions below 500 cm^{-1})

5 Summary

We have implemented a novel algorithm based on an existing alternating least squares algorithm for transforming high-dimensional potential energy surfaces into a canonical tensor form for efficient use in the Heidelberg MCTDH software package. The algorithm allows for the treatment of much higher dimensional surfaces than possible so far. This was demonstrated by calculating the linear absorption spectrum of the Eigen cation. For this calculation, the 33-dimensional potential energy surface as well as the dipole moment surfaces were fitted. The fit was obtained by utilizing up to 19200 CPU cores in parallel on the HLRS Hazel Hen system.

The absorption spectrum of the Eigen cation obtained with help of these calculations is in very good agreement with the experimental results. All major features in the experimental spectrum have been obtained in the model calculation. Especially the prominent peaks associated with the water wagging, O-O-distance and proton transfer are obtained at the correct relative position and with the correct shape.

In summary, our developments and calculations allow for the first time a detailed and complete quantum mechanical modeling of the Eigen cation. Our calculations will help to obtain a deeper understanding of the complex dynamical processes that occur upon triggering a proton transfer and which are involved in the broad proton transfer spectral feature. This subject is still under debate and further investigations are underway.

References

1. H.-D. Meyer, U. Manthe, L.S. Cederbaum, Chem. Phys. Lett. **165**, 73 (1990)
2. U. Manthe, H.-D. Meyer, L.S. Cederbaum, J. Chem. Phys. **97**, 3199 (1992)
3. M.H. Beck, A. Jäckle, G.A. Worth, H.-D. Meyer, Phys. Rep. **324**, 1 (2000)
4. H.-D. Meyer, G.A. Worth, Theor. Chem. Acc. **109**, 251 (2003)
5. *Multidimensional Quantum Dynamics: MCTDH Theory and Applications*, ed. by H.-D. Meyer, F. Gatti, G.A. Worth (Wiley-VCH, Weinheim, 2009)
6. H.-D. Meyer, WIREs: Comput. Mol. Sci. **2**, 351 (2012)
7. H. Wang, J. Phys. Chem. A **119**, 7951 (2015)
8. O. Vendrell, H.-D. Meyer, J. Chem. Phys. 134, 044135 (2011)
9. H. Wang, M. Thoss, J. Chem. Phys. 131, 024114 (2009)
10. U. Manthe, J. Chem. Phys. 128, 164116 (2008)
11. H. Wang, M. Thoss, J. Chem. Phys. **119**, 1289 (2003)
12. A. Jäckle, H.-D. Meyer, J. Chem. Phys. **104**, 7974 (1996)
13. A. Jäckle, H.-D. Meyer, J. Chem. Phys. **109**, 3772 (1998)
14. D. Peláez, H.-D. Meyer, J. Chem. Phys. 138, 014108 (2013)
15. F. Otto, J. Chem. Phys. 140, 014106 (2014)
16. M. Schröder, H.-D. Meyer, J. Chem. Phys. 147, 064105 (2017)
17. F. Otto, Y.-C. Chiang, D. Peláez, Chem. Phys. **509**, 116 (2018)
18. F.L. Hitchcock, J. Math. Phys. 6, 164189 (1927)
19. M. Schröder, J. Chem. Phys. 152, 024108 (2020)
20. N. Metropolis et al., J. Chem. Phys. **21**, 1087 (1953)
21. P. Comon, X. Luciani, A.L.F. de Almeida, J. Chemom. **23**, 393 (2009)
22. T.K. Esser et al., J. Chem. Phys. Lett. **9**, 798 (2018)
23. C.T. Wolke et al., Science **354**, 1131 (2016)
24. M. Ndong et al., J. Chem. Phys. 136, 034107 (2012)

Ab Initio Molecular Dynamics Simulation of Vibrational Energy Relaxation at the Solid/Liquid Interface

Charge Defects at the Fluorite/Water Interface Allow Very Fast Intermolecular Vibrational Energy Transfer

Dominika Lesnicki and Marialore Sulpizi

Abstract The water/fluorite interface is of relevance to diverse industrial, environmental, and medical applications. In this contribution we review some of our recent results on the dynamics of water in contact with the solid calcium fluoride at low pH, where localised charge can develop upon fluorite dissolution. We use *ab initio* molecular dynamics simulations, including the full electronic structure, to simulate the vibrational energy relaxation and to quantify the heterogeneity of the interfacial water molecules. We find that strongly hydrogen-bonded OH groups display very rapid spectral diffusion and vibrational relaxation; for weakly H-bonded OD groups, the dynamics is instead much slower. Detailed analysis of the simulations reveals the molecular origin of energy transport through the local hydrogen-bond network. In particular, we find that the water molecules in the adsorbed layer, whose orientation is pinned by the localised charge defects, can exchange vibrational energy using just half a solvation shell, thanks to the strong dipole-dipole alignment between H-bond donor and acceptor.

Keywords *Ab initio* molecular dynamics · Vibrational energy relaxation · Non-equilibrium MD · 2D-Sum Frequency Generation · Solid/liquid interfaces

1 Introduction

In this contribution we review some of our recent results on the structural and dynamical characterisation of calcium fluorite/water interfaces obtained from *ab initio*

D. Lesnicki · M. Sulpizi (✉)
Johannes Gutenberg University, Staudingerweg 7, 55099 Mainz, Germany
e-mail: sulpizi@uni-mainz.de

Present Address:
D. Lesnicki
Sorbonne University, Paris, France

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_6

molecular dynamics simulations. Atomistic molecular dynamics (MD) simulations, and in particular *ab initio* simulations can help to gain insight into the interfaces between mineral surfaces and water, which is the key to understand a variety of natural phenomena, including e.g., the weathering of rocks and corrosions, as well as to advance technological/industrial applications [25, 26]. The specific case of the calcium fluorite (CaF_2)/water interface, which we present here, is of relevance to industrial, environmental, and medical applications, e.g., for understanding fluorine dissolution in drinking water.

At their meeting point, the interface, the properties of both liquid and solid are expected to be different from those observed in the bulk. This is due e.g. in the case of liquid water, to the fact that the hydrogen bonding network of water is interrupted and strongly influenced by the solid surface. As a consequence of the different structure, also the dynamics of the liquid is strongly affected and depending on the specific interface, may result to be slower or faster than in bulk water. A selective tool to investigate water dynamics and to address vibrational energy relaxation at interfaces is time-resolved (tr) and two-dimensional (2D) sum frequency generation spectroscopy (SFG), which has been extensively used in the last ten years to address the water surface and obtain its molecular picture [2, 11, 12, 22, 31]. In tr- and 2D-SFG, a fraction of the water molecules (about 10%) is excited with an intense infrared pulse and the subsequent energy relaxation can be selectively monitored with the SFG probe, a combination of a broadband infrared pulse (which is resonant with the molecular vibrations) and a narrow-band visible pulse. The strength of the SFG spectroscopy resides in the fact that it selectively probes only the water molecules at the interfaces, as its selection rule is such that no signal will be obtained from centro-symmetric media like bulk water [14]. At the interface, where the symmetry is broken, a signal is instead present and vibrational spectra can be experimentally recorded. In the case of the water surface (or in other words the water-air interface), an energy relaxation slower than in bulk water has been observed [24], as water can reorient faster than in the bulk [10], thanks to a reduced hydrogen-bonding at the interface.

tr-SFG has been used also employed to characterise the water/mineral interface [3–6, 20, 28, 29]. One of the most characterized mineral/water interfaces is certainly the silica/water interface, possibly the most abundant in nature. The first femtosecond tr-SFG experiments on the water/silica interface was performed by McGuire and Shen [20]. They found a fast 300 fs time constant for the bleach relaxation. Further, subsequent experiments by Eric Borguet and coworkers explored the effect of pH and ionic strength on the vibrational dynamics of both silica/water and alumina/water interfaces. In the case of the silica/water interface, the relaxation time scales of interfacial water were found to strongly depend on the local hydrogen bond network, the local surface charge, and finally by the isotopic dilution [4–6]. On the other hand for the alumina/water interface, the vibrational relaxation dynamics of H-bonded water was found to be insensitive to surface charge and ionic strength. Additional investigation also showed that the relaxation dynamics of water at the charged alumina interface is faster than for bulk water [28, 29]. If tr-SFG can provide information on the vibrational lifetimes, 2D-SFG can additionally also provide the spectral diffusion

dynamics, and therefore further information on structure, structural dynamics, and interfacial energy transfer dynamics [31].

Although MD simulations can, in principle, permit to calculate the experimentally observed 2D-SFG spectra using the appropriate response functions [21], the overall computational cost to obtain converged spectra is still prohibitive, in particular if one aims to use electronic structure-based approaches. Instead of directly compute response functions, an alternative approach can be to directly simulate the vibrational excitation and the subsequent vibrational energy relaxation [16]. The results, which we are going to present in this review are based on such an approach. Here we therefore use non-equilibrium molecular dynamics simulations in order to calculate both vibrational time relaxation and spectral diffusion, as well as to provide a molecular interpretation of the experiments [16, 18]. Our approach includes an atomistic description of the interface, where the interatomic potentials are determined by the full electronic structure at the density functional theory (DFT) level. In our previous report for HRLS [17] we have shown how such models have been able to successfully reproduce the static SFG spectra of the CaF_2/water interface over a wide range of pH [13, 17]. Here we go beyond the discussion of the static vibrational spectra in terms of the structure of water at the CaF_2 interface. Instead we show how such description of the interfaces at the nanoscale is also accurate to reproduce the time-resolved spectra and to address dynamical properties.

2 Computational Approach

2.1 Simulations Set-Up

To obtain molecular-level details of the relaxation mechanism, we use simulations of the vibrational relaxation using *ab initio* molecular dynamics simulations according to the method reported in Ref. [16, 18]. Our starting point is the model for the low pH interface with 0.64 vacancies/nm^2 (1 vacancy per surface) [13]. The interface between CaF_2 (111) and water is composed of 88 water molecules and 60 formula units of CaF_2 contained in a 11.59 Å × 13.38 Å × 34.0 Å cell periodically repeated in the (x, y, z) directions. The thickness of water slabs is around 20 Å along the z-axis, which is a reasonable compromise between the need to achieve bulk-like properties far from the surface and the computational cost. All the simulations have been carried out with the package CP2K/Quickstep [30], consisting in Born-Oppenheimer MD (BOMD) BLYP [1, 15] electronic representation including Grimme (D3) correction for dispersion [8], GTH pseudo-potentials [7, 9], a combined Plane-Wave (280 Ry density cutoff) and TZV2P basis sets (Fig. 1).

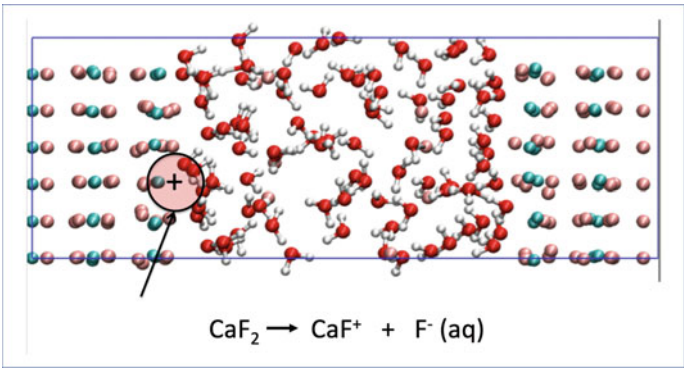


Fig. 1 Model for the low pH interface with 0.64 vacancies/nm^2 (1 vacancy per surface) [13]. The vacancy is created by the dissolution of surface fluoride ions according to the formula shown here. The interface between CaF_2 (111) and water is composed of 88 water molecules and 60 formula units of CaF_2 contained in a 11.59 Å × 13.38 Å × 34.0 Å cell periodically repeated in the (x, y, z) directions.

2.2 *Vibrational Energy Excitation and Relaxation from Non-equilibrium Trajectories*

For the analysis of the vibrational energy relaxation we used the same approach presented in our previous work [16, 18]. We start from an equilibrated trajectory in the NVT ensemble with the Nose-Hoover thermostat where a time-step of 0.5 fs for integrating the equation of motion was used. From this trajectory 82 snapshots were extracted every 5 ps in order to ensure that velocities were decorrelated. These snapshots were used as the initial coordinates and velocities for the non-equilibrium AIMD runs. The average temperature in the NVT AIMD simulation was 330 K in order to reproduce the water liquid structure at 1 g/cm^3 [27]. We then ran 2 ps AIMD simulations with a time-step of 0.1 fs for the excited and not excited case in the NVE ensemble. Excitations were performed for the water molecules located within 3Å away from the surfaces (composed of fluorine), leading to a total of 552 excitations. We excite each single water molecule such that the temperature of the simulation box is increased by 1.5 K. This is close to the increment in the temperature of the sample which is measured in the experiments and also ensures that the excitation is not strongly perturbing, or even disrupting, the local hydrogen bond network. Figure 2 shows the frequency distribution obtained from the simulations. The extra kinetic energy, added to a given molecule, drives, instantaneously, the system out of the equilibrium and the subsequent relaxation process can be followed within NVE trajectories using descriptors, which we introduced in Ref. [16, 18] and which permit to follow the excess energy redistribution among the given modes. The time evolution of the excess vibrational energy can be obtained as difference between the power spectrum calculated in the vibrationally excited (or non equilibrium) state and that calculated in the ground (or equilibrium) state, according to:

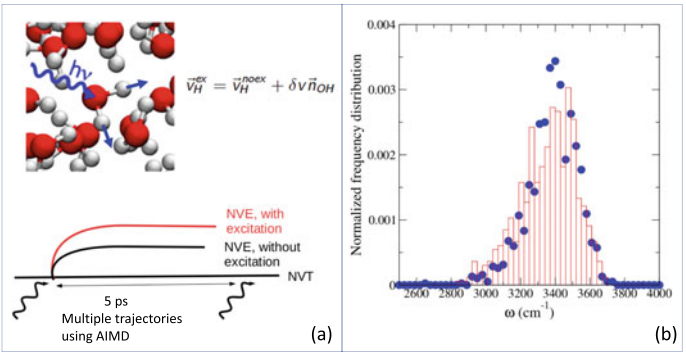


Fig. 2 (a) Scheme of the simulation procedure. Along an equilibrium trajectory excitations are created adding a velocity component along a given normal mode. (b) Normalised distribution of the stretch frequencies of the water located in the first layer (red) [18] and in the bulk (blue) [16] at t = 0.1 ps. (Color figure online)

$$I(t) = \int_0^\infty P^{ex}(\omega, t)\ d\omega - \int_0^\infty P^{gs}(\omega, t)\ d\omega \tag{1}$$

where

$$P(\omega, t) = \int_t^{t+\Delta t} \left\langle \sum_i \mathbf{v}_i(t)\mathbf{v}_i(t+\tau) \right\rangle e^{-i\omega\tau}\ d\tau. \tag{2}$$

is the Fourier transform of the velocity-velocity autocorrelation function calculated at a given time t and $\mathbf{v}_i(t)$ is the velocity of atom i at time t and Δt the time window over which the correlation function is computed.

2.3 *Spectral Diffusion from Equilibrium and Non-equilibrium Simulations*

The spectral diffusion can be directly obtained in the non-equilibrium simulations from the time decay of the frequency-frequency correlation function (FFCF) defined by

$$C_\omega(t) = \langle \delta\omega(t)\delta\omega(0) \rangle \,/\, \langle \delta\omega(0)^2 \rangle \tag{3}$$

where $\delta\omega(t) = \omega_{max}(t) - \bar{\omega}_{max}(t)$ is the fluctuation from the average frequency at time t. $C_\omega(t)$ can be calculated both in equilibrium, as well as in non-equilibrium simulations, namely following the vibrational excitation. The results of the frequency-frequency correlation are shown in Fig. 3 for both the equilibrium case (black dots) and the non equilibrium case (red dots). For both cases a fast exponential decay of 140 fs and 100 fs, respectively, are found in agreement with the experiments and

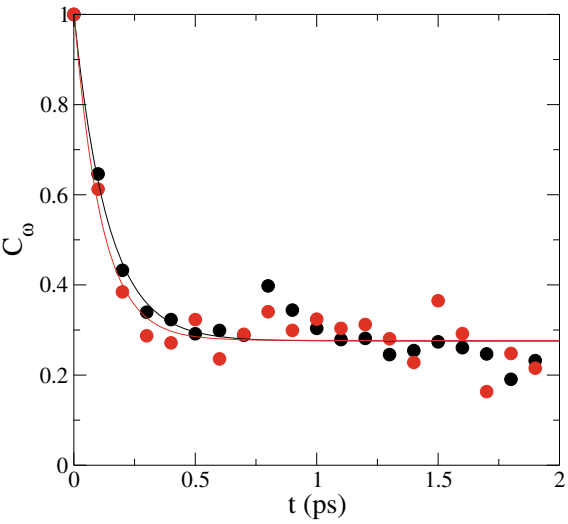


Fig. 3 Frequency-frequency correlation function as function of time for both the equilibrium case (black dots) and the non equilibrium case (red dots). A single exponential fit provides a lifetime of 140 fs and 100 fs for the equilibrium (black curve) and non equilibrium (red curve) cases, respectively. (Color figure online)

the value reported for bulk water [16, 24]. For such calculations only the water molecules in the interfacial layer with a thickness of 3.5Å are included, which are those contributing most to the SFG signal, according to our previous analysis [13].

2.4 Mean Square Displacement

To quantify the dynamical properties of the water molecules in different environments, interfacial water and bulk, we have computed the mean square displacement (MSD). The MSD is defined from the trajectories $\mathbf{r}_i(t)$ of the diffusing particles, labelled with the index i, in terms of the distance from their initial position $\mathbf{r}_i(0)$:

$$\left\langle \Delta\mathbf{r}(t)^2 \right\rangle = \frac{1}{N_w} \left\langle \sum_{i=1}^{N_w} |\mathbf{r}_i(t) - \mathbf{r}_i(0)|^2 \right\rangle \tag{4}$$

where N_w is the number of water molecules considered. For the calculation of the MSD, we have used a sliding window of 10 ps over 50 ps trajectories of water at CaF_2 interface [13] and bulk water [16, 18]. For the former trajectory, we have selected water molecules up to 3.5 Å from the interface. The diffusion coefficient, D, was then computed using the Einstein relation:

$$D = \frac{1}{6} \lim_{t \to +\infty} \frac{d}{dt} \left\langle \Delta\mathbf{r}(t)^2 \right\rangle \tag{5}$$

3 Results

3.1 A Very Fast Spectral Diffusion

In our recent publication [18] we have reported on a joint computational and experimental effort to understand the water dynamics at the fluorite water interface. In Ref. [18] the 2D experimental spectrum for the CaF_2 interface is reported. A first interesting feature to notice is that at early delay times it shows a small tilt along the diagonal between 2200 and 2500 cm^{-1}, indicating a very fast vibrational energy transfer. With increasing delay time, the bleach becomes more horizontal due to additional spectral diffusion. The time evolution of the slope permits to extract information on the spectral diffusion, namely the time evolution of the instantaneous frequency of the molecules in the systems. In the case of the fluorite/water interface at low pH the slope decays with an exponential behaviour with a time constant below 100 fs [18], which is comparable to the decay observed in bulk water [16, 24], and substantially faster than what has been observed at the water/air interface [31] and water-positively charged lipid interfaces [19] with D_2O as sub-phase.

The spectral diffusion can also be obtained from the *ab initio* molecular dynamics (AIMD) simulations presented in Ref. [13, 18] by calculating the frequency-frequency correlation functions (Fig. 3). For such a calculation only the interfacial layer is used with a thickness of 3.5 Å, which is the layer contributing most to the SFG signal, according to our previous analysis [13]. A fast exponential decay of the order of 100 fs is found. This is in agreement with the experimental results and it is also similar to the value obtained from the simulations of bulk water in Ref. [16]. It is quite remarkable that the energy transfer dynamics at the interface is comparable to that in bulk water, as in the bulk the density of OH (or, equivalently, OD) groups is much higher. This very fast interfacial energy transfer, therefore, therefore suggests and enhanced intermolecular coupling at the interface, which could be a result of the surface-induced alignment of water molecules. We will come back to this point when discussing the energy relaxation in the subsequent paragraph.

3.2 Vibrational Energy Relaxation: The Role of Charge Defect on the Surface

Besides the time evolution of the slope, the time-dependent 2D plots also provide information about energy relaxation. The details on the experimental results are reported in our recent publication [18], here we would like to focus on the computational side of such study.

Starting from an equilibrated trajectory, we selectively excite the water molecules in the first water layer in contact with the fluorite surface. Every single excitation is obtained by adding, to a given water molecule, excess kinetic energy with the symmetry of the stretching mode. After the vibrational excitation we can use NVE

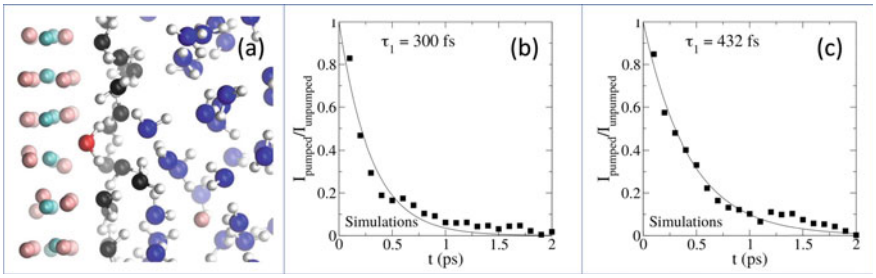


Fig. 4 (a) Snapshot from the simulation: the water molecules highlighted in black are those excited in the non-equilibrium simulations. (b, c) Time evolution of the excess energy of the OH stretch obtained from simulations at low pH (square) for the water ensemble with a pump frequency below 2515 cm^{-1} (b) and above 2515 cm^{-1} (c) normalised by the initial value and its exponential fit (plain line).

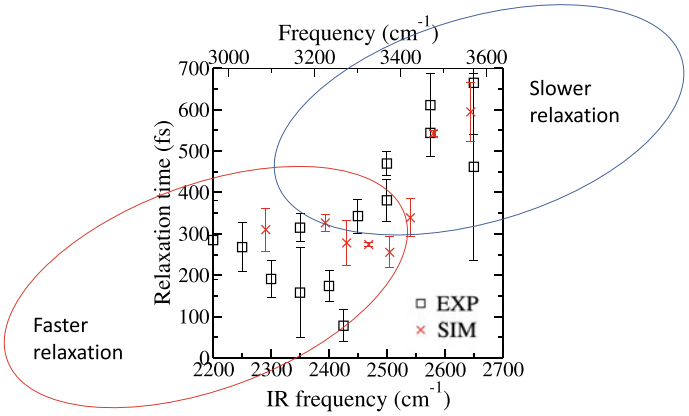


Fig. 5 Vibrational relaxation time as function of frequency (black: experimental data; red: simulation). For both experiments and simulations faster relaxation times at low frequency and slower relaxation times at higher frequencies are observed [18]. Data are from Ref. [16, 18]. (Color figure online)

trajectories, to follow how the excess energy added to the system leaves the original excited water molecule and is eventually redistributed over the system. As an example, the relaxation of vibrational energy out of the excited stretching state for excitation frequencies below and above 2515 cm^{-1} are reported in Fig. 4 (panels (b) and (c) for an average excitation frequency of 2400 cm^{-1} and 2590 cm^{-1}, respectively).

In Fig. 5 we report the frequency-dependent relaxation times from the simulations (red crosses) compared with those from experiments (black squares). For both experiments and simulations faster relaxation times at low frequency and slower relaxation times at higher frequencies are observed. As discussed in our previous work [18] we should be careful in comparing the timescales from simulations and experiments as

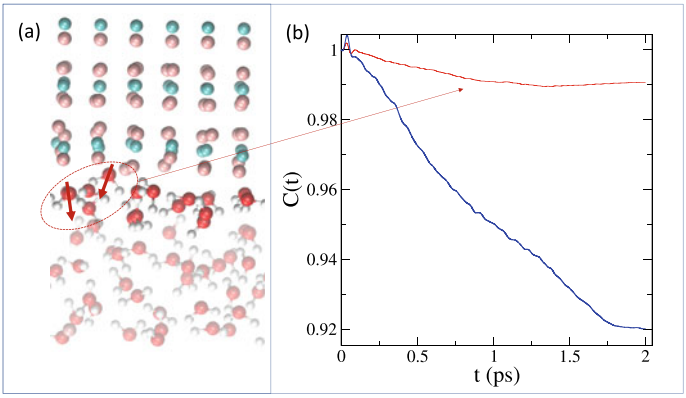


Fig. 6 (a) A snapshot of the model used in the simulations. The water molecules in the adsorbed layer and excited in the simulation of the vibrational relaxation are highlighted in full color, while the rest is depicted as transparent. (b) Time evolution of the dipole-dipole correlation function for the adsorbed water in the first layer (red) and for water in the bulk (blue). Data are from Ref. [16, 18]. (Color figure online)

the simulations are performed in H_2O and the experiments in D_2O, as one would expect slightly slower dynamics in the latter case.

To understand the molecular origin of the observed energy relaxation we analyse the simulations in more detail. In particular, it is interesting to discuss the structure of the adsorbed layer. Such a layer is highlighted in Fig. 6(a), while the rest of the water molecules are rendered as transparent. At a difference with respect to bulk water, the structure of the interfacial layer is well-ordered, as a result of the water interaction with the positive charge defects localised in correspondence of the fluoride vacancy.

As we already pointed out in our previous work [13] such order extends over 4–5 Å. This was indeed inferred from the calculation from the convergence of the static spectrum, in particular of the Im $\chi^{(2)}$ part, as function of the increasing probing thickness [13]. We should, however, note here that the high computational cost of electronic structure based methods imposes severe limitations on the size of the accessible models. In this respect, our model is expected to capture the contribution to the spectra originating from the Stern layer (possibly the major contribution here), but it cannot account for the full diffuse layer, which is expected to extend over a few nanometers thickness. To capture this second contribution a mixed resolution approach including both the Stern layer in the full electronic structure details, as well as the diffuse layer, possibly at the force field level, could be considered in the future. As also mentioned in our previous work [18], the experimental data indicate that the contribution of the diffuse layer is small. Close inspection of the interfacial layer reveals that the water molecules directly pinned by the surface defects do not move for the entire length of the simulations, as e.g. can be also shown in the analysis of the root mean square displacement (Fig. 7). The water molecules in the interfacial layer diffuse slower, D = 2.197 10^{-10}, than the water molecules in bulk water, D

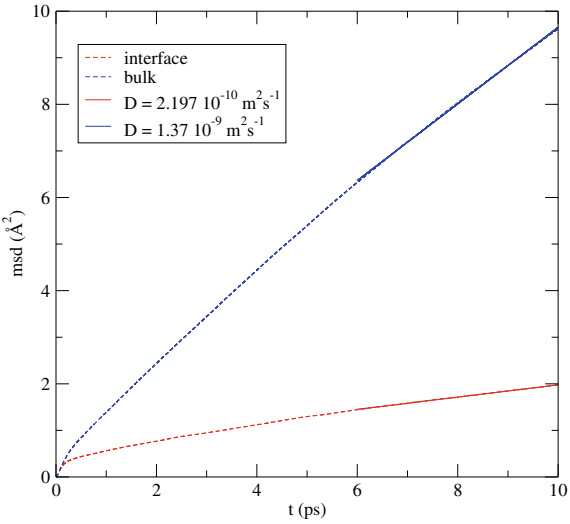


Fig. 7 Mean square displacement of the water molecules in the first layer (red) and for water in the bulk (blue). The plain lines display the linear fit used to compute the diffusion coefficient. (Color figure online)

= 1.37 10^{-9} (see Fig. 7). Such "frozen" water molecules are also characterised by only half of a solvation shell, as they only donate two hydrogen bonds while not accepting any. The order imposed by the charge defects also extends to the other water molecules which reside in the interfacial layer, and which, on average, form only 2.45 H-bonds (with 62% donors and 38% acceptors), which is one H-bond less when compared to bulk water, where molecules form on average 3.48 H-bonds (with equal acceptor and donor contributions). If comparing to bulk water, the interfacial water molecules have an incomplete first solvation shell. Interestingly, though, the interconnecting H-bond network at the interface is also much more stable, and the intra-layer H-bond dynamics much slower. This can be appreciated by calculating the correlation function between the dipole moment of each water molecule in the layer and its neighbouring, H-bonded water molecules. Such a curve is reported in red in Fig. 6b). The correlation function is close to one when two molecules do not re-orient with respect to each other, namely they maintain their reciprocal dipole orientation as a function of time. This happens when the H-bond remains intact for the entire length of the trajectory. The correlation function decays to zero, if the reciprocal orientation is lost, or in other words, if the H-bond is broken. Looking at Fig. 6b) we notice that for the water in the first adsorbed layer on the surface, the dipole-dipole correlation function (red curve) decays very slowly, while for bulk water (blue curve) it decays much faster.

The simulations permits to unveil a quite interesting behaviour of the interfacial water and in particular of the layer directly in contact with the fluorite surface. Indeed, the interfacial water molecules present a strongly ordered and an asymmetric H-bond network. With respect to bulk water, they miss almost half of their solvation shell. However, the increased order, produced by the localised positive charges on

the surface, is somehow able to compensate for the missing H-bonds, enabling fast vibrational energy transfer, despite the lower effective water density. It is interesting to compare the behaviour of this adsorbed layer with that of the water molecules in ice Ih. In case of ice strong, ordered H-bonds bonds can lead to relaxation time constants as short as 80 fs [23], however the relaxation is possible through a complete first solvation shell. Here, at the fluorite interface strong, ordered H-bonds can transfer fast, but they only use half a solvation shell. As a consequence the resulting relaxation time constants are still fast, as in bulk water.

Acknowledgements This work was supported by the Deutsche Forschungsgemeinschaft (DFG) TRR146, project A4. All the calculation were performed on the supercomputer of the High Performance Computing Center (HLRS) of Stuttgart (grant 2DSFG).

References

1. A.D. Becke, Density-functional exchange-energy approximation with correct asymptotic behavior. Phys. Rev. A **38**, 3098–3100 (1988). https://doi.org/10.1103/PhysRevA.38.3098. https://link.aps.org/doi/10.1103/PhysRevA.38.3098
2. M. Bonn, Y. Nagata, E.H.G. Backus, Molecular structure and dynamics of water at the water–air interface studied with surface-specific vibrational spectroscopy. Angewandte Chemie Int. Ed. **54**(19), 5560–5576. https://doi.org/10.1002/anie.201411188. https://onlinelibrary.wiley.com/doi/abs/10.1002/anie.201411188
3. J.D. Cyran, M.A. Donovan, D. Vollmer, F. Siro Brigiano, S. Pezzotti, D.R. Galimberti, M.P. Gaigeot, M. Bonn, E.H.G. Backus, Molecular hydrophobicity at a macroscopically hydrophilic surface. Proc. Natl. Acad. Sci. **116**(5), 1520–1525 (2019). https://doi.org/10.1073/pnas.1819000116. https://www.pnas.org/content/116/5/1520
4. A. Eftekhari-Bafrooei, E. Borguet, Effect of surface charge on the vibrational dynamics of interfacial water. J. Am. Chem. Soc. **131**(34), 12034–12035 (2009). https://doi.org/10.1021/ja903340e. PMID: 19663486
5. A. Eftekhari-Bafrooei, E. Borguet, Effect of hydrogen-bond strength on the vibrational relaxation of interfacial water. J. Am. Chem. Soc. **132**(11), 3756–3761 (2010). https://doi.org/10.1021/ja907745r. PMID: 20184315
6. A. Eftekhari-Bafrooei, E. Borguet, Effect of electric fields on the ultrafast vibrational relaxation of water at a charged solid-liquid interface as probed by vibrational sum frequency generation. J. Phys. Chem. Lett. **2**(12), 1353–1358 (2011). https://doi.org/10.1021/jz200194e
7. S. Goedecker, M. Teter, J. Hutter, Separable dual-space gaussian pseudopotentials. Phys. Rev. B **54**, 1703–1710 (1996). https://doi.org/10.1103/PhysRevB.54.1703. https://link.aps.org/doi/10.1103/PhysRevB.54.1703
8. S. Grimme, J. Antony, S. Ehrlich, H. Krieg, A consistent and accurate ab initio parametrization of density functional dispersion correction (dft-d) for the 94 elements h-pu. J. Chem. Phys. **132**(15), 154104 (2010). https://doi.org/10.1063/1.3382344
9. C. Hartwigsen, S. Goedecker, J. Hutter, Relativistic separable dual-space Gaussian pseudopotentials from H to Rn. Phys. Rev. B **58**, 3641–3662 (1998). https://doi.org/10.1103/PhysRevB.58.3641. https://link.aps.org/doi/10.1103/PhysRevB.58.3641
10. C.S. Hsieh, R.K. Campen, A.C. Vila Verde, P. Bolhuis, H.K. Nienhuys, M. Bonn, Ultrafast reorientation of dangling oh groups at the air-water interface using femtosecond vibrational spectroscopy. Phys. Rev. Lett. **107**, 116102 (2011). https://doi.org/10.1103/PhysRevLett.107.116102. https://link.aps.org/doi/10.1103/PhysRevLett.107.116102

11. C.S. Hsieh, M. Okuno, J. Hunger, E.H.G. Backus, Y. Nagata, M. Bonn, Aqueous heterogeneity at the air/water interface revealed by 2d-hd-sfg spectroscopy. Angewandte Chemie Int. Ed. **53**(31), 8146–8149. https://doi.org/10.1002/anie.201402566. https://onlinelibrary.wiley.com/doi/abs/10.1002/anie.201402566
12. K.I. Inoue, T. Ishiyama, S. Nihonyanagi, S. Yamaguchi, A. Morita, T. Tahara, Efficient spectral diffusion at the air/water interface revealed by femtosecond time-resolved heterodyne-detected vibrational sum frequency generation spectroscopy. J. Phys. Chem. Lett. **7**(10), 1811–1815 (2016). https://doi.org/10.1021/acs.jpclett.6b00701. PMID: 27120559
13. R. Khatib, E.H.G. Backus, M. Bonn, M.J. Perez-Haro, M.P. Gaigeot, M. Sulpizi, Water orientation and hydrogen-bond structure at the fluorite/water interface. Sci. Rep. **6**, 24287 (2016). https://doi.org/10.1038/srep24287
14. J.P. Kraack, P. Hamm, Surface-sensitive and surface-specific ultrafast two-dimensional vibrational spectroscopy. Chem. Rev. **117**(16), 10623–10664 (2017). https://doi.org/10.1021/acs.chemrev.6b00437. PMID: 28830147
15. C. Lee, W. Yang, R.G. Parr, Development of the colle-salvetti correlation-energy formula into a functional of the electron density. Phys. Rev. B **37**, 785–789 (1988). https://doi.org/10.1103/PhysRevB.37.785. https://link.aps.org/doi/10.1103/PhysRevB.37.785
16. D. Lesnicki, M. Sulpizi, A microscopic interpretation of pump-probe vibrational spectroscopy using ab initio molecular dynamics. J. Phys. Chem. B **122**(25), 6604–6609 (2018). https://doi.org/10.1021/acs.jpcb.8b04159. PMID: 29799755
17. D. Lesnicki, M. Sulpizi, Enhanced acid dissociation at the solid/liquid interface, in *High Performance Computing in Science and Engineering 2018*. ed. by W.E. Nagel, D.H. Kröner, M.M. Resch (Springer International Publishing, Cham, 2019), pp. 141–151
18. D. Lesnicki, Z. Zhang, M. Bonn, M. Sulpizi, E.H. Backus, Surface charges at the CaF2 water interface allow very fast intermolecular vibrational energy transfer. Angewandte Chemie Int. Ed. **n/a**(n/a). https://doi.org/10.1002/anie.202004686. https://onlinelibrary.wiley.com/doi/abs/10.1002/anie.202004686
19. R.A. Livingstone, Z. Zhang, L. Piatkowski, H.J. Bakker, J. Hunger, M. Bonn, E.H.G. Backus, Water in contact with a cationic lipid exhibits bulklike vibrational dynamics. J. Phys. Chem. B **120**(38), 10069–10078 (2016). https://doi.org/10.1021/acs.jpcb.6b07085. PMID: 27564997
20. J.A. McGuire, Y.R. Shen, Ultrafast vibrational dynamics at water interfaces. Science **313**(5795), 1945–1948 (2006). https://doi.org/10.1126/science.1131536. http://science.sciencemag.org/content/313/5795/1945
21. S. Mukamel, *Principles of Nonlinear Optical Spectroscopy*, vol. 6 (Oxford University Press, Oxford, 1995)
22. S. Nihonyanagi, S. Yamaguchi, T. Tahara, Ultrafast dynamics at water interfaces studied by vibrational sum frequency generation spectroscopy. Chem. Rev. **117**(16), 10665–10693 (2017). https://doi.org/10.1021/acs.chemrev.6b00728. PMID: 28378588
23. F. Perakis, P. Hamm, Two-dimensional infrared spectroscopy of neat ice ih. Phys. Chem. Chem. Phys. **14**, 6250–6256 (2012). https://doi.org/10.1039/C2CP23710E. http://dx.doi.org/10.1039/C2CP23710E
24. S.T. van der Post, C.S. Hsieh, M. Okuno, Y. Nagata, H.J. Bakker, M. Bonn, J. Hunger, Strong frequency dependence of vibrational relaxation in bulk and surface water reveals sub-picosecond structural heterogeneity. Nat. Commun. **6**, 8384 (2015). https://doi.org/10.1038/ncomms9384
25. A. Putnis, Why mineral interfaces matter. Science **343**(6178), 1441–1442 (2014). https://doi.org/10.1126/science.1250884. http://science.sciencemag.org/content/343/6178/1441
26. C.V. Putnis, E. Ruiz-Agudo, The mineral-water interface: Where minerals react with the environment. Elements **9**(3), 177–182 (2013). https://doi.org/10.2113/gselements.9.3.177. http://dx.doi.org/10.2113/gselements.9.3.177
27. J. Schmidt, J. VandeVondele, I.F.W. Kuo, D. Sebastiani, J.I. Siepmann, J. Hutter, C.J. Mundy, Isobaricisothermal molecular dynamics simulations utilizing density functional theory: An assessment of the structure and density of water at near-ambient conditions. J. Phys. Chem. B **113**(35), 11959–11964 (2009). https://doi.org/10.1021/jp901990u. PMID: 19663399

28. A. Tuladhar, S. Dewan, J.D. Kubicki, E. Borguet, Spectroscopy and ultrafast vibrational dynamics of strongly hydrogen bonded oh species at the -al2o3(1120)/h2o interface. J. Phys. Chem. C **120**(29), 16153–16161 (2016). https://doi.org/10.1021/acs.jpcc.5b12486
29. A. Tuladhar, S.M. Piontek, E. Borguet, Insights on interfacial structure, dynamics, and proton transfer from ultrafast vibrational sum frequency generation spectroscopy of the alumina(0001)/water interface. J. Phys. Chem. C **121**(9), 5168–5177 (2017). https://doi.org/10.1021/acs.jpcc.7b00499
30. J. VandeVondele, M. Krack, F. Mohamed, M. Parrinello, T. Chassaing, J. Hutter, Quickstep: fast and accurate density functional calculations using a mixed gaussian and plane waves approach. Comput. Phys. Commun. 167(2), 103–128 (2005). https://doi.org/10.1016/j.cpc.2004.12.014. http://www.sciencedirect.com/science/article/pii/S0010465505000615
31. Z. Zhang, L. Piatkowski, H. Bakker, M. Bonn, Ultrafast vibrational energy transfer at the water/air interface revealed by two-dimensional surface vibrational spectroscopy. Nat. Chem. **3**, 888–893 (2011)

Computational Study of Confinement Effects in Molecular Heterogeneous Catalysis

Hamzeh Kraus, Julia Rybka, Ulrich Tallarek, and Niels Hansen

Abstract The properties of fluid mixtures composed of solvent, reactant and product molecules within functionalized mesoporous materials and the local composition of reacting species around the catalytically active complex anchored covalently inside the inner pore space are important factors determining the activity of catalytic reactions in confined geometries. Here, a computational approach based on classical molecular dynamics simulations is presented, allowing to investigate a catalyst immobilization strategy for a ring-closing metathesis reaction ahead of its experimental realization. The simulations show that using epoxide groups as immobilization sites leads to a spatial distribution of reactant and product molecules that appears to be counterproductive to a desired confinement effect.

Keywords Mesopores · Confinement · Catalysis

1 Introduction

Recent advances in designing mesoporous silica materials with surface-anchored functional groups have shown promising potential for applications with respect to the activity and selectivity in molecular heterogeneous catalysis [20, 39, 51]. For example, by immobilizing a Rh-complex on SBA-15 particles, a rate enhancement for 1-octene hydroformulation was observed compared to the homogeneous analogue, which was explained by the suppression of the formation of inactive forms of the catalyst inside the pores [40]. Other examples include improved selectivity for olefin or cyclooctane metathesis reactions carried out in mesoporous silica materials [9, 45] or synergetic effects between multifunctional groups anchored in close proximity in confined spaces [54]. Selectivity and conversion of reactions can be affected

H. Kraus · N. Hansen (✉)
Institute of Thermodynamics and Thermal Process Engineering, University of Stuttgart, 70569 Stuttgart, Germany
e-mail: hansen@itt.uni-stuttgart.de

J. Rybka · U. Tallarek
Department of Chemistry, Philipps-Universität Marburg, 35032 Marburg, Germany

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_7

due to controlling the factors that influence the spatial distribution and the mobility of reactants and products including molecular size, shape, configuration, degree of confinement, pore topology, strength of adsorption on pore walls, and the possibility of hydrogen bonding between reactants or/and products. Instrumental for the application and rational design of functionalized mesoporous materials is the existence of a model for predicting how pore dimension and surface functionalization influence the properties of the fluid under confinement [8]. Fluids confined in mesoporous materials present a particular challenge to empirical tuning, as it is difficult to experimentally probe properties such as solvent composition and phase behavior within the mesopores. Mass transport of molecular compounds through porous solids is a decisive step in molecular heterogeneous catalysis. It is a multi-scale, hierarchical phenomenon: Effective diffusion through the macro-mesoporous material is influenced by parameters such as grain boundaries and particle packing on the macropore scale ($>\mu$m), as well as by factors such as particle size and connectivity of pores on the mesopore scale (>10 nm, $<\mu$m). More importantly, meso-scale diffusion and macro-scale diffusion are first and foremost determined directly by processes on the molecular scale (<10 nm), which depend on numerous factors like pore-size, interactions of the diffusing species with the solid surfaces and with the solvent [29]. For reacting systems transport of the reactants to and products from the catalytically active complex is decisive as well as the interaction of the various species with the solvent and the functional groups on the surface. Therefore, there is a growing interest in computational modeling studies to provide a fundamental molecular-level description of confinement effects [17, 23].

An important step towards this goal is to gain a detailed knowledge of physical processes at the solid-liquid interface in porous materials. For chromatographic interfaces molecular dynamics simulations have been used to study both reversed phase liquid chromatography (RPLC) and hydrophilic interaction liquid chromatography (HILIC), in particular (i) the structure and dynamics of the bonded phase and its interface with the mobile phase, (ii) the interactions of analytes with the bonded phase, and (iii) the retention mechanism for different analytes [34, 41]. By simulations it was shown, for example, that retention into octyl (C8) phases is best described as an adsorption process, while for octadecyl (C18) phases both adsorption and partition play a role for nonpolar analytes, whereas adsorption is always the major mechanism for analyte molecules with polar groups that lead to an amphiphilic character. A recent study investigated the surface diffusion of four typical aromatic hydrocarbon analytes in RPLC through molecular dynamics simulations in a slit-pore RPLC model consisting of a silica-supported end-capped, C18 stationary phase and a 70/30 (v/v) water/acetonitrile mobile phase. The results show that the lateral (surface-parallel) diffusive mobility of the analytes goes through a maximum in the acetonitrile ditch, an acetonitrile-rich border layer around the terminal part of the bonded-phase chains [48].

In the present work we study confinement effects using the ring-closing metathesis reaction of dec-9-en-1-yl undec-10-enoate (C_{11-10}) to the monocyclisation product oxacycloicos-11-ene-2-one (see Fig. 1) in benzene solvent over a 2nd-generation Grubbs-Hoveyda-type catalyst immobilized inside an epoxy-functionalized [10, 19,

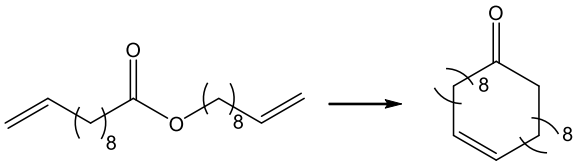


Fig. 1 Reactant and product molecules for an exemplary ring-closing metathesis reaction considered in the present work.

42, 47] mesopore as example. Such a reaction is relevant for the production of macrocyclic compounds that play an important role in pharmaceutical chemistry.

2 Methodology

2.1 Computational Model of a Functionalized Mesopore

Initial Pore. The cylindrical mesopore model consisted of a 6.02 nm diameter pore carved through the (111) face of a β-cristobalite block (9.11 nm × 8.77 nm × 10.08 nm ($x \times y \times z$)) along the z-direction using the in-house python package PoreMS [30, 32], following procedures reported previously [13, 41], resulting in 5.52 silanol groups nm^{-2} on the inner pore surface. The cylindrical pore was flanked by two solvent reservoirs with the outer surfaces bearing 4.98 silanol groups nm^{-2}. Force-field parameters for Si, O, and H atoms of the silica surface were taken from Gulmen and Thompson [22, 49]. Examples of generated pores are shown in Fig. 2.

Surface Functionalization. All molecules used for functionalizing the surface are shown in Fig. 3. Their topologies need to be supplied for single (silicon atom with one unsaturated oxygen) and geminal (silicon atom with two unsaturated oxygens) binding sites. These topologies were obtained using the Antechamber topology builder based on coordinate files generated by the molecule builder functionality of the PoreMS package. However, the topology builder does not include all atom types needed such as silicon and ruthenium. Therefore, slightly altered structures were passed to Antechamber, saturating the surface silicon atom and replacing unknown atoms with similar ones such as carbon for silicon. In this way, an initial topology could be generated, and the unknown parameters, or further needed ones, were added manually.

When functionalizing the surface, the user can define the allocation, either as a number of molecules, for example for a specific number of catalysts, or as a percentage of the surface. If a geminal oxygen is selected for functionalization, the other oxygen is saturated with hydrogen.

Catalyst Anchored at the Inner Pore Wall. The catalytically active center is identical to the one studied previously [57], the only difference is the longer linker used to attach the catalyst to the pore surface. The catalyst was modeled in its precursor form with an undissociated Ru-O bond due to the high degree of uncertainty involved

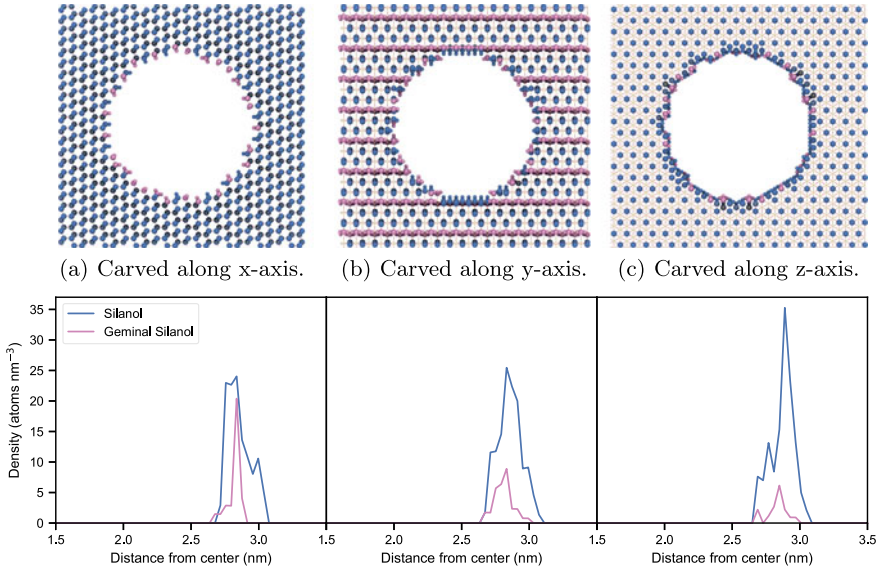


Fig. 2 The top panels show the frontal views on the constructed pore systems. The color code used is: red, oxygen atoms; yellow, silicon atoms; blue, single silanol groups; pink, geminal silanol groups. The bottom panels display the number density of the oxygen atoms of single silanol (blue) and geminal silanol (pink) molecules. A pore of diameter 6 nm was carved in a 10 nm silica block through the x-axis (left), y-axis (center) and z-axis (right). (Color figure online)

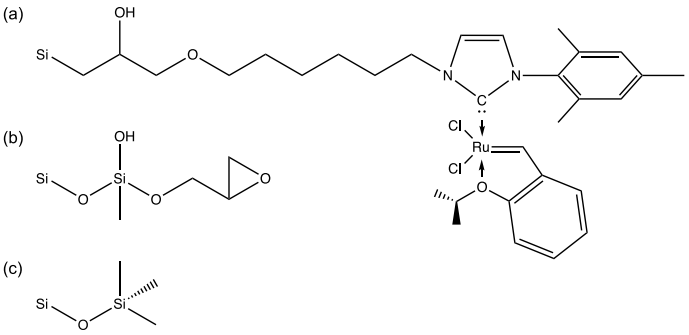


Fig. 3 Chemical structure of molecules attached to the surface. (a) Grubbs-Hoveyda-type catalyst; (b) epoxide group; (c) Trimethyl-silyl group.

in describing the interaction of the activated form of the catalyst with reactants and products in the context of classical force fields and the focus of the simulations on confinement effects. A DFT optimization was performed with the TURBOMOLE program [1, 3] using five different functionals (B3LYP [6, 50], PW6B95 [55], M06 [56], M06-L [56], PBE0 [44]) and two different basis sets (def2-SVP and def2-TZVP) [53]. Except for B3LYP these functionals and basis sets were among the

recommended ones for the calculation of structural parameters in Ru-complexes based on a comparison to MP2 calculations [33]. For both tested basis sets the PW6B95 functional had the smallest deviation from the X-ray structure [57] based on a comparison of selected bond lengths around the ruthenium atom.

To proceed with the parametrization most force field parameters were taken from the GAFF force field [52] while for the Ru-complex geometric parameters calculated in the present work were used and combined with force constants reported by Ahmadi et al. [4]. Due to the rather rigid geometry of the precursor state the actual values of the force constants e.g. for keeping the aromatic rings planar are not expected to impact the simulation results significantly.

To model the nonbonded interactions Lennard-Jones parameters were taken from the GAFF force field except for ruthenium which was described using a 12-6 potential fitted to the 9-6 form used by Ahmadi [4]. Compared to other molecular mechanics parameters reported for ruthenium [5, 18] the energy parameter reported by Ahmadi et al. [4] is rather large. Therefore, the interactions between the catalytic complex and a methane probe molecule was calculated for representative configurations using dispersion-corrected DFT [21] and compared to the force field calculations for different Lennard-Jones parameters of Ru. Since ruthenium is rather shielded the actual LJ-energy parameter hardly affects the interaction with a methane probe molecule. The strength of the interaction is in reasonable agreement with dispersion corrected DFT results.

Partial charges of the catalytic complex were extracted from the electron densities calculated with the PW6B95 functional and a def2-TZVP basis set using the DDEC [36–38] method implemented in the DDEC6 program [36]. This approach was chosen due to the compatibility of the DDEC partial charges with the GAFF force field as demonstrated previously [7].

Configuration files of the equilibrated pore systems, corresponding topologies and simulation parameter files are provided via the Data Repository of the University of Stuttgart (DaRUS) [31].

2.2 Molecular Dynamics Simulations

Simulated Systems. In order to analyze the confinement effect, two different pore sizes were simulated each with pure substrate, product and a mixture of both, with benzene as a solvent. In an NPT-simulation, the target density of 6.38 molecules/nm^3 for the solvent benzene and 0.04 molecules/nm^3 for the solutes substrate and product was determined, with which the pore simulation box was filled. All pores had a surface allocation of 5.71 μmol/m^2 inside the pore (epoxide) and 3.72 μmol/m^2 outside the pore (TMS). Table 1 shows the simulated system properties. The system containing both product and substrate are filled equally towards the same total density.

Figure 4 exemplarily shows a simulation system containing a 6 nm pore modified with epoxide containing groups inside a mesopore at a grafting density of 3.44 groups/nm^2 and TMS groups on the outside at a grafting density of 2.24 groups/nm^2.

Table 1 Pore diameter d, dimensions of the silica block D and number of solvent (benzene) and solute (substrate and/or product) molecules used in the two simulated system sizes.

d (nm)	D (nm)	N_{solvent}	N_{solute}
6	[9.10, 8.77, 9.66]	6333	44
9	[12.14, 12.28, 9.66]	12897	85

The inner pore surface additionally bears two catalyst groups which stick out from the epoxide-modified surface (see front view on the cylindrical pore in panel a)). Substrate and product molecules dissolved in benzene are distributed between the solvent reservoirs outside of the pore and inside the cylindrical mesopore.

Simulation Parameters. Molecular dynamics simulations were carried out with the GROMACS 2016.5 program package [2, 26] compiled in single precision. All simulations were performed under minimum image periodic boundary conditions in a rectangular simulation box with fixed dimensions containing the silica mesopore between two bulk phase reservoirs. Initial velocities were randomly assigned according to a Maxwell-Boltzmann distribution. The desired bulk phase concentration of all species was achieved iteratively in a series of short simulations at constant volume by adding or removing molecules. After a 100 ns equilibration period a production simulation was run at constant volume and constant temperature at 300 K for 200 ns. The equations of motion were integrated with the leap frog scheme [27] applying a time-step of 1 fs. The length of bonds involving a hydrogen atom were kept fixed using LINCS [24, 25] with an order of 4. The number of iterations to correct for rotational lengthening in LINCS was set to 2. The temperature was kept close to its reference temperature by application of the Nosé-Hoover thermostat [28, 43] with a relaxation time of 1.0 ps. Short range electrostatic and Lennard-Jones interactions were treated within a cut-off radius of 1.4 nm. Analytic dispersion corrections were applied for energy and pressure. The particle-mesh-Ewald (PME) method [15, 16] was used for treating long-range electrostatic interactions.

Simulations were run on the ForHLR I Cluster. A short benchmark for 1 ps showed an optimal ratio between efficiency and simulation time at core number of 200 (10 nodes) with 9 ns/24 h. The speedup at this core number is 8 compared to 20 cores (1 node). Beyond this point the slope of the speedup curve is almost halved.

Analysis. Density profiles were calculated from the local number density $\rho_{n,i}$ by counting the number of molecules N_i in volume slices V_i. Inside the pore this is done by creating a radial slicing such that the sub volumes are calculated by

$$V_i^{\text{pore}} = \pi \cdot z_{\text{pore}}(r_i^2 - r_{i-1}^2) \tag{1}$$

with pore length z_{pore} and radius r_i of sub volume i. This yields the radial number density inside the pore as

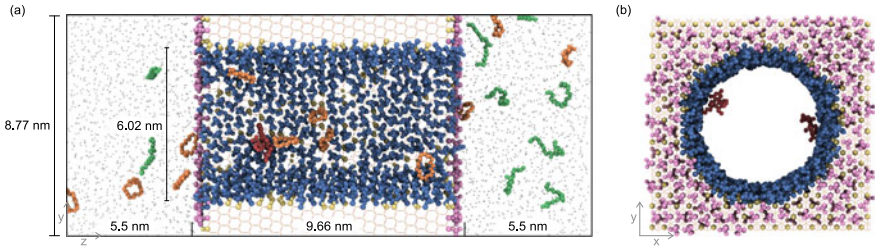


Fig. 4 Cylindrical catalytic mesopore model generated with PoreMS. (a) Side view of the simulation box indicating the length of the central silica block and the solvent reservoirs. (b) Front view, of the pierced silica block containing the 6 nm pore. The chemistry of the exterior surface is based on the (111) face of β-cristobalite silica. The exterior planar and interior curved surfaces are covered with randomly distributed TMS and epoxide groups, respectively. Two organometallic catalyst groups are attached to the interior surface in point symmetry with respect to the pore centre. Colour code: Si atoms, yellow line; O atoms, red line; Epoxide groups, blue; TMS groups, magenta; catalyst, red; residual surface silanol groups, yellow; Benzene, grey beads; substrate, green; product, orange. (Color figure online)

$$\rho_{n,i}^{\text{pore}} = \frac{N_i}{V_i^{\text{pore}}} = \frac{1}{\pi \cdot z_{\text{pore}}} \frac{N_i}{r_i^2 - r_{i-1}^2}. \tag{2}$$

Outside the pore, the sub volumes are calculated along the distance orthogonal to the front surface. If the pore was carved along the z-axis it follows

$$V_j^{\text{out}} = z_j(x \cdot y - \pi \cdot r^2) \tag{3}$$

with box width x, box height y, pore radius r and slice width z_j of slice j. Thus, the number density along the z-coordinate is

$$\rho_{n,j}^{\text{out}} = \frac{N_j}{V_j^{\text{out}}} = \frac{1}{x \cdot y - \pi \cdot r^2} \frac{N_j}{z_j}. \tag{4}$$

Note that *outside* refers to the reservoirs of the simulation box. Therefore, the slices add up to the reservoir length z_{res}. Due to the symmetry of the system the profiles from both reservoirs were averaged. Since the outside density refers to the density measured from the outside surface, it does not contain the cylindrical extension of the pore inside the reservoirs. Finally, the mass density is calculated by

$$\rho = \frac{M}{N_A} \rho_n \tag{5}$$

with molar mass M and Avogadro constant N_A. For a more detailed representation, density profiles are generated for particular atoms of a molecule. For the substrate and product this includes the chain oxygen-atoms and the carbon atoms that form the double bond in the reaction.

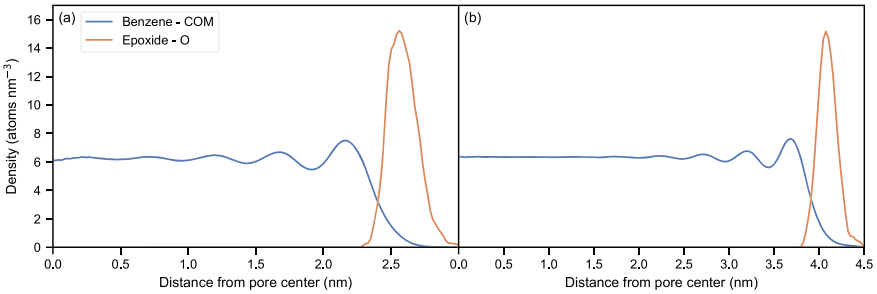


Fig. 5 Radial number density profiles of benzene center of mass and the oxygen atoms of the epoxide group at 300 K in the (a) 6 nm pore and (b) the 9 nm pore.

The local diffusion coefficient in the direction parallel to the pore axis was calculated following an approach by Liu et al. [35]. To obtain a progression of the diffusion coefficient along the radial coordinate, volume slices similar to the density calculation were implemented. In each slice j the mean square displacement $\langle z^2(r_j, t)\rangle$ was repeatedly recorded for molecules that remained in the specified space interval for the whole observation time of 20 ps, giving the diffusion coefficient from

$$\langle z^2(r_j, t)\rangle = D_{||}(r_j)t \tag{6}$$

if $\langle z^2(r_j, t)\rangle$ could be described by a straight line in the time interval 4 ps to 16 ps.

3 Results

3.1 Distribution of Solvent, Reactant and Product Molecules Inside the Mesopore

Figure 5 shows the density profiles of the solvent benzene as well as of the epoxide groups within the inner pore volume for the two studied pore sizes in case of an equimolar mixture of reactant and product molecules in the system. The epoxide groups show well-defined density peaks close to the silica surface as expected. The distribution function of benzene displays a typical shape with four well defined maxima as has been observed also in other simulation studies on silica surfaces [11, 12, 14]. However, the density maximum close to the pore wall is less pronounced compared to pure silanol surfaces. For the 9 nm pore the uniform density is reached at around 2.5 nm from the pore center while for the 6 nm pore small density oscillations remain even close to the pore center. The number densities at the pore center are slightly lower than the bulk values at the same temperature. Compared to the bulk phase density distribution function (not shown) the fluid structure in the pore is longer

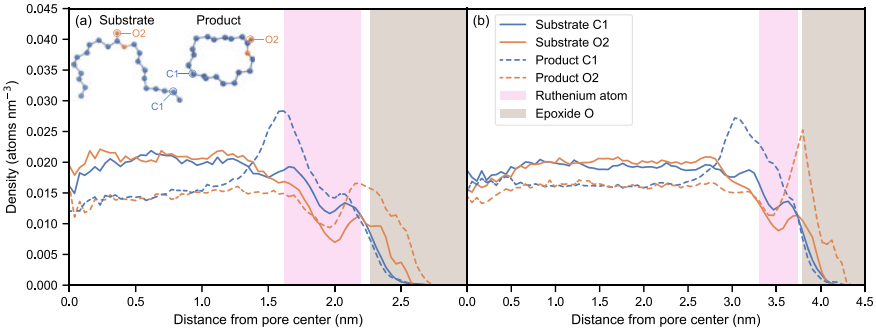


Fig. 6 Radial number density profiles of the carbonyl oxygen (red) and vinylic carbon (blue) from simulations at 300 K containing an equimolar amount of substrate (solid line) and product (dashed line) molecules in cylindrical pores of 6 nm (a) and 9 nm (b) diameter. The shaded areas indicate the configurational space accessible by the ruthenium atom of the catalyst (pink) and the oxygen atom of the epoxide-group (brown). (Color figure online)

ranged since it shows a higher number of coordination shells, a consequence of the fluid-wall interactions. The density profiles of benzene and the epoxide groups for the other two studied system compositions are very similar and therefore not shown.

Figure 6 shows the radial number density profiles of particular atoms of the substrate and product molecules for both the 6 nm pore and the 9 nm pore analyzed in the simulation containing an equimolar mixture of substrate and product molecules. The shaded areas present in all panels indicate the configurational space accessible by the ruthenium atom of the catalyst. The substrate molecules are relatively uniformly distributed over the pore cross section and show a depletion towards the pore wall. By contrast, the product molecules show an enrichment close to the surface which is particularly pronounced for the 6 nm pore. This pattern is retained in case of simulations containing either substrate or product molecules, respectively (data not shown). This suggests that an epoxide functionalization is unfavorable for this reaction because local enrichment of product molecules near the catalytic centers increases the risk of back-biting reactions.

3.2 Mobility of Solvent, Reactant and Product Molecules Inside the Mesopore

Figure 7 shows the self-diffusion coefficient of benzene parallel to the pore axis as function of the radial coordinate. The mobility decreases as expected monotonically from the bulk region to practically zero at the surface. While in the 6 nm pore no distinct plateau is formed towards the pore center, such a plateau reaches up to 2.5 nm in the 9 nm pore. The self-diffusion coefficients obtained at the center of the pore are consistent with values calculated from independent bulk phase

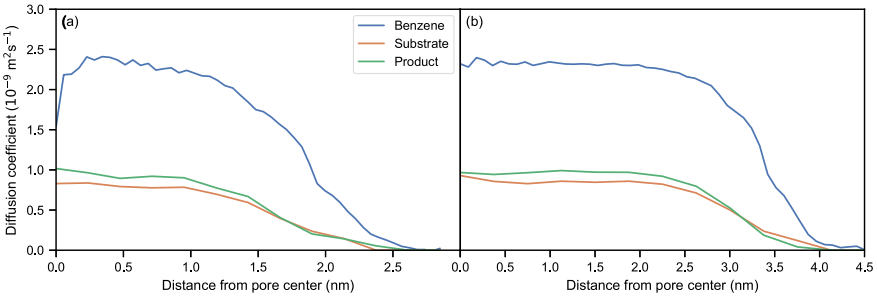


Fig. 7 Radial dependence of the axial self-diffusion coefficients at 300 K of benzene as well as the substrate and product molecules, respectively, in (a) the 6 nm pore and (b) the 9 nm pore.

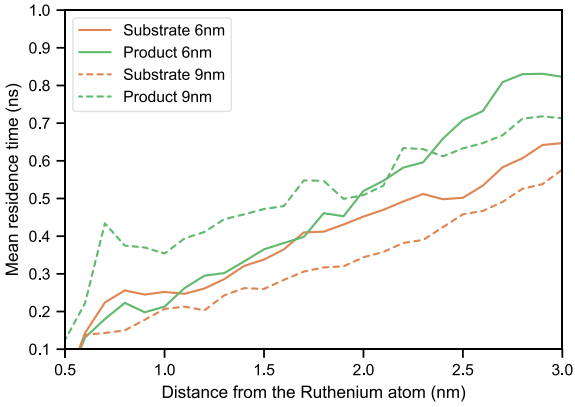


Fig. 8 Mean residence time of the centers of mass of substrate and product molecules around the ruthenium atom as function of the distance from the latter. Only those molecules contributed to the average which stayed at least for 0.1 ns within the specified distance.

simulations of mixtures of benzene with either reactant or product molecules, respectively, resulting in $D_{\text{benz}} = 2.3 \times 10^{-9}\,\text{m}^2/\text{s}$. Also shown in Fig. 7 are the self-diffusion coefficients of the reactant and product molecules parallel to the pore axis. The qualitative behavior is similar to that of benzene. The numerical values obtained at the pore center are consistent with those resulting from independent bulk phase simulations of mixtures of benzene with either reactant or product molecules, i.e. $D_{\text{reac}} = 0.9 \pm 0.1 \times 10^{-9}\,\text{m}^2/\text{s}$ and $D_{\text{prod}} = 1.0 \pm 0.1 \times 10^{-9}\,\text{m}^2/\text{s}$.

Finally, the residence times of substrate and product molecules around the ruthenium atom have been analyzed. Figure 8 shows that substrate and product molecules have very similar residence times close to the catalytically active center. This shows that in combination with the higher density of product molecules towards the pore wall a risk of back-biting reactions is rather high with the epoxide-grafted surface.

4 Discussion

Local enrichment of product molecules near the catalytic centers indicates that the epoxide surface increases the risk of oligomerization and is thus not optimal for the metathesis reaction. Apart from calculating the self-diffusion coefficients a further important factor for the characterization of transport through a porous material is the diffusion hindrance factor, which quantifies the degree to which diffusion is hindered depending on λ, the ratio of solute size to mean pore size [46]. The correlation between the hindrance factor and λ was established from simulations in the physically-reconstructed mesopore space of three macro-mesoporous silica monoliths by Reich et al. [46]. For the systems studied here, λ is 0.22 in a 9 nm and 0.37 in a 6 nm pore. This leads to hindrance factors of 0.7 in a 9 nm pore and 0.45 in a 6 nm pore showing that below 6 nm pore diameter severe diffusion limitation is to be expected.

5 Conclusions

A computational approach is presented based on the generation of functionalized silica model pores for molecular simulation studies of confinement effects in molecular heterogeneous catalysis. The method was used to study an immobilization strategy for the catalyst based on an epoxide functionalization of the inner pore surface. The spatial distribution of substrate and product molecules revealed that the epoxide surface leads to a depletion of reactants and enrichment of product molecules close to the surface which may be counterproductive to a desired confinement effect. Therefore, molecular simulation studies allow to support the selection of suitable reaction systems ahead of expensive and time-consuming experiments.

Acknowledgments This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), project ID 358283783 – SFB 1333, by the Ministry of Science, Research and Arts and the Universities of the State of Baden-Württemberg, Germany, and the Federal Ministry of Education and Research of Germany, project number 16FDM008. Most of this work was performed on the supercomputer ForHLR funded by the Ministry of Science, Research and the Arts Baden-Württemberg and by the Federal Ministry of Education and Research. The authors also appreciate support by the High Performance and Cloud Computing Group at the Zentrum für Datenverarbeitung of the University of Tübingen, the state of Baden-Württemberg through bwHPC and the German Research Foundation (DFG) through grant no INST 37/935-1 FUGG. Fruitful discussions with Sereina Riniker, Patrick Bleiziffer, Felix Ziegler and Michael Buchmeiser are gratefully acknowledged.

References

1. TURBOMOLE v.7.2.1 2017, A development of University of Karlsruhe and Forschungszentrum Karlsruhe GmbH, 1989–2007, TURBOMOLE GmbH, since 2007. http://www.turbomole.com
2. M.J. Abraham, T. Murtola, R. Schulz, S. Páll, J.C. Smith, B. Hess, E. Lindahl, GROMACS: high performance molecular simulations through multi-level parallelism from laptops to supercomputers. SoftwareX **1–2**, 19–25 (2015)
3. R. Ahlrichs, M. Bär, M. Häser, H. Horn, C. Kölmel, Electronic structure calculations on workstation computers: the program system turbomole. Chem. Phys. Lett. **162**, 165–169 (1989)
4. A. Ahmadi, C. McBride, J.J. Freire, A. Kajetanowicz, J. Czaban, K. Grela, Force field parametrization and molecular dynamics simulation of flexible POSS-linked (NHC; phosphine) Ru catalytic complexes. J. Phys. Chem. A **115**, 12017–12024 (2011)
5. N.L. Allinger, X. Zhou, J. Bergsma, Molecular mechanics parameters. J. Mol. Struct. (Theochem) **312**, 69–83 (1994)
6. A.D. Becke, Density-functional thermochemistry. III. The role of exact exchange. J. Chem. Phys. **98**, 5648–5652 (1993)
7. P. Bleiziffer, K. Schaller, S. Riniker, Machine learning of partial charges derived from high-quality quantum-mechanical calculations. J. Chem. Inf. Model. **58**, 579–590 (2018)
8. A. Boţan, F.J. Ulm, R.J.M. Pellenq, B. Coasne, Bottom-up model of adsorption and transport in multiscale porous media. Phys. Rev. E **91**, 032133 (2015)
9. M. Bru, R. Dehn, J.H. Teles, S. Deuerlein, M. Danz, I.B. Müller, M. Limbach, Ruthenium carbenes supported on mesoporous silicas as highly active and selective hybrid catalysts for olefin metathesis reactions under continuous flow. Chem. Eur. J. **19**, 11661–11671 (2013)
10. N. Calin, A. Galarneau, T. Cacciaguerra, R. Denoyel, F. Fajula, Epoxy-functionalized large-pore SBA-15 and KIT-6 as affinity chromatography supports. C. R. Chimie **13**(1–2), 199–206 (2010)
11. B. Coasne, C. Alba-Simionesco, F. Audonnet, G. Dosseh, K.E. Gubbins, Molecular simulation of the adsorption and structure of benzene confined in mesoporous silica. Adsorption **13**, 485–490 (2007)
12. B. Coasne, C. Alba-Simionesco, F. Audonnet, G. Dosseh, K.E. Gubbins, Adsorption and structure of benzene on silica surfaces and in nanopores. Langmuir **25**, 10648–10659 (2009)
13. B. Coasne, F. Di Renzo, A. Galarneau, R.J.M. Pellenq, Adsorption of simple fluid on silica surface and nanopore: effect of surface chemistry and pore shape. Langmuir **24**, 7285–7293 (2008)
14. B. Coasne, J.T. Fourkas, Structure and dynamics of benzene confined in silica nanopores. J. Phys. Chem. C **115**, 15471–15479 (2011)
15. T. Darden, D. York, L. Pedersen, Particle mesh Ewald: an N log(N) method for Ewald sums in large systems. J. Chem. Phys. **98**, 10089–10092 (1993)
16. U. Essmann, L. Perera, M. Berkowitz, T. Darden, H. Lee, L. Pedersen, A smooth particle mesh Ewald method. J. Chem. Phys. **103**, 8577–8593 (1995)
17. A. García, I.I. Slowing, J.W. Evans, Pore diameter dependence of catalytic activity: p-nitrobenzaldehyde conversion to an aldol product in amine-functionalized mesoporous silica. J. Chem. Phys. **149**, 024101 (2018)
18. S. Geremia, M. Calligaris, Stereochemical and conformational features of ruthenium sulfoxide complexes: a molecular mechanics approach. J. Chem. Soc. Dalton Trans. **9**, 1541–1547 (1997)
19. P. Gholamzadeh, G.M. Ziarani, A. Badiei, Immobilization of lipases onto the SBA-15 mesoporous silica. Biocatal. Biotransfor. **35**(3), 131–150 (2017)
20. F. Goettmann, C. Sanchez, How does confinement affect the catalytic activity of mesoporous materials. J. Mater. Chem. **17**, 24–30 (2007)
21. S. Grimme, J. Antony, S. Ehrlich, H. Krieg, A consistent and accurate ab initio parametrization of density functional dispersion correction (DFT-D) for the 94 elements H-Pu. J. Chem. Phys. **132**, 154104 (2010)

22. T.S. Gulmen, W.H. Thompson, Testing a two-state model of nanoconfined liquids: conformational equilibrium of ethylene glycol in amorphous silica pores. Langmuir **22**, 10919–10923 (2006)
23. N. Hansen, F.J. Keil, Multiscale modeling of reaction and diffusion in zeolites: from the molecular level to the reactor. Soft Mater. **10**, 179–201 (2012)
24. B. Hess, P-LINCS: a parallel linear constraint solver for molecular simulation. J. Chem. Theory Comput. **4**, 116–122 (2008)
25. B. Hess, H. Bekker, H.J.C. Berendsen, J.G.E.M. Fraaije, LINCS: a linear constraint solver for molecular simulation. J. Comput. Chem. **18**, 1463–1472 (1997)
26. B. Hess, C. Kutzner, D. van der Spoel, E. Lindahl, GROMACS 4: algorithms for highly efficient, load-balanced, and scalable molecular simulation. J. Chem. Theory Comput. **4**, 435–447 (2008)
27. R.W. Hockney, The potential calculation and some applications. Methods Comput. Phys. **9**, 136–211 (1970)
28. W. Hoover, Canonical dynamics: equilibrium phase-space distributions. Phys. Rev. A **31**, 1695–1697 (1985)
29. P. Huber, Soft matter in hard confinement: phase transition thermodynamics, structure, texture, diffusion and flow in nanoporous media. J. Phys.: Condens. Matter **27**, 103102 (2015)
30. H. Kraus, N. Hansen, PoreMS: v0.2.0 (2020). https://doi.org/10.5281/zenodo.3984865
31. H. Kraus, J. Rybka, Supplementary material for 'Computational study of confinement effects in molecular heterogeneous catalysis' (2020). https://doi.org/10.18419/darus-488
32. H. Kraus, J. Rybka, A. Höltzel, N. Trebel, U. Tallarek, N. Hansen, PoreMS: a software tool for generating silica pore models with user-defined surface functionalization and pore dimensions. Mol. Simul. **47**, 306–316 (2021)
33. A.D. Kulkarni, D.G. Truhlar, Performance of density functional theory and Møller-Plesset second-order perturbation theory for structural parameters in complexes of Ru. J. Chem. Theory Comput. **7**, 2325–2332 (2011)
34. R.K. Lindsey, J.L. Rafferty, B.L. Eggimann, J.I. Siepmann, M.R. Schure, Molecular simulation studies of reversed-phase liquid chromatography. J. Chromatogr. A **1287**, 60–82 (2013)
35. P. Liu, E. Harder, B.J. Berne, On the calculation of diffusion coefficients in confined fluids and interfaces with an application to the liquid-vapor interface of water. J. Phys. Chem. B **108**, 6595–6602 (2004)
36. T.A. Manz, N.G. Limas, Introducing DDEC6 atomic population analysis: Part 1. Charge partitioning theory and methodology. RSC Adv. **6**, 47771–47801 (2016)
37. T.A. Manz, D.S. Sholl, Chemically meaningful atomic charges that reproduce the electrostatic potential in periodic and nonperiodic materials. J. Chem. Theory Comput. **6**, 2455–2468 (2010)
38. T.A. Manz, D.S. Sholl, Improved atoms-in-molecule charge partitioning functional for simultaneously reproducing the electrostatic potential and chemical state in periodic and nonperiodic materials. J. Chem. Theory Comput. **8**, 2844–2867 (2012)
39. E.L. Margelesfky, R.K. Zeidan, M.E. Davis, Cooperative catalysis by silica-supported organic functional groups. Chem. Soc. Rev. **37**, 1118–1126 (2008)
40. F. Marras, J. Wang, M.O. Coppens, J.N.H. Reek, Ordered mesoporous materials as solid supports for rhodium-diphosphine catalysts with remarkable hydroformulation activity. Chem. Comm. **46**, 6587–6589 (2010)
41. S.M. Melnikov, A. Höltzel, A. Seidel-Morgenstern, U. Tallarek, A molecular dynamics study on the partitioning mechanism in hydrophilic interaction chromatography. Angew. Chem. Int. Ed. **51**, 6251–6254 (2012)
42. M. Mohammadi, M.A. As'habi, P. Salehi, M. Yousefi, M. Nazari, J. Brask, Immobilization of laccase on epoxy-functionalized silica and its application in biodegradation of phenolic compounds. Int. J. Biol. Macromol. **109**, 443–447 (2018)
43. S. Nosé, A molecular dynamics method for simulations in the canonical ensemble. Mol. Phys. **52**, 255–268 (1984)
44. J.P. Perdew, M. Ernzerhof, K. Burke, Rationale for mixing exact exchange with density functional approximations. J. Chem. Phys. **105**, 9982–9985 (1996)

45. E. Pump, Z. Cao, M.K. Samantaray, A. Bendjeriou-Sedjerari, L. Cavallo, J.M. Basset, Exploiting confinement effects to tune selectivity in cyclooctane metathesis. ACS Catal. **7**, 6581–6586 (2017)
46. S.J. Reich, A. Svidrytski, D. Hlushkou, D. Stoeckel, C. Kübel, A. Höltzel, U. Tallarek, Hindrance factor expression for diffusion in random mesoporous adsorbents obtained from pore-scale simulations in physical reconstructions. Ind. Eng. Chem. Res. **57**, 3031–3042 (2018)
47. G. Renard, M. Mureseanu, A. Galarneau, D.A. Lerner, D. Brunel, Immobilisation of a biological chelate in porous mesostructured silica for selective metal removal from wastewater and its recovery. New J. Chem. **29**(7), 912–918 (2005)
48. J. Rybka, A. Höltzel, U. Tallarek, Surface diffusion of aromatic hydrocarbon analytes in reversed-phase liquid chromatography. J. Phys. Chem. C **121**, 17907–17920 (2017)
49. K.G. Steenbergen, J.L. Kern, Z. Wang, W.H. Thompson, B.B. Laird, Tunability of gas-expanded liquids under confinement: phase equilibrium and transport properties of ethylene-expanded methanol in mesoporous silica. J. Phys. Chem. C **120**(9), 5010–5019 (2016)
50. P.J. Stephens, F.J. Devlin, C.F. Chabalowski, M.J. Frisch, Ab initio calculation of vibrational absorption and circular dichroism spectra using density functional force fields. J. Phys. Chem. **98**, 11623–11627 (1994)
51. J. Wang, Q. Ma, Y. Wang, Z. Li, Z. Li, Q. Yuan, New insights into the structure-performance relationships of mesoporous materials in analytical science. Chem. Soc. Rev. **47**, 8766–8803 (2018)
52. J. Wang, W. Wang, P.A. Kollman, D.A. Case, Automatic atom type and bond type perception in molecular mechanical calculations. J. Mol. Graph. Model. **25**, 247–260 (2006)
53. F. Weigend, R. Ahlrichs, Balanced basis sets of split valence, triple zeta valence and quadruple zeta valence quality for H to Rn: design and assessment of accuracy. Phys. Chem. Chem. Phys. **7**, 3297–3305 (2005)
54. C. Yu, J. He, Synergetic catalytic effects in confined spaces. Chem. Comm. **48**, 4933–4940 (2012)
55. Y. Zhao, D.G. Truhlar, Design of density functionals that are broadly accurate for thermochemistry, thermochemical kinetics, and nonbonded interactions. J. Phys. Chem. A **109**, 5656–5667 (2005)
56. Y. Zhao, D.G. Truhlar, The M06 suite of density functionals for main group thermochemistry, thermochemical kinetics, noncovalent interactions, excited states, and transition elements: two new functionals and systematic testing of four M06-class functionals and 12 other functionals. Theor. Chem. Acc. **120**, 215–241 (2008)
57. F. Ziegler, J. Teske, I. Elser, M. Dyballa, W. Frey, H. Kraus, N. Hansen, J. Rybka, U. Tallarek, M.R. Buchmeiser, Olefin metathesis in confined geometries: a biomimetic approach toward selective macrocyclization. J. Am. Chem. Soc. **141**, 19014–19022 (2019)

Simulation of Nonequilibrium Spin Dynamics in Quantum Dots Subjected to Periodic Laser Pulses

Philipp Schering, Philipp W. Scherer, and Götz S. Uhrig

Abstract Large-scale simulations of the spin dynamics in quantum dots subjected to trains of periodic laser pulses enable us to describe and understand related experiments. By comparing the data for different models to experimental results, we gain an improved understanding of the relevant physical mechanisms. Using sophisticated numerical approaches and an efficient implementation combined with extrapolation arguments, nonequilibrium stationary states are reached for parameter ranges close to the ones in real experiments. With the help of high performance computing, we can tune the experimental parameters to guide future experimental research. Importantly, our simulations reveal the possibility of resonant spin amplification in Faraday geometry, i.e., when a longitudinal magnetic field is applied to the quantum dots.

1 Introduction

A localized electronic spin in a semiconductor quantum dot (QD) is considered a promising candidate for the realization of quantum bits [1], which are at the very basis of any quantum information processing [2]. Such an electronic spin in a QD loses its coherence due to its interaction with the bath of nuclear spins of the surrounding isotopes in III-V semiconductors. The number of substantially coupled nuclear spins is very large of the order of 10^4 to 10^6 [3].

Considerable effort has been invested in the experimental investigation of the spin dynamics in semiconductor nanostructures and the possibilities to manipulate it [3–5]. It is particularly interesting that ensembles of QDs can be manipulated as well. They can be made to respond coherently by subjecting them to long periodic trains of laser pulses while applying a transverse magnetic field [6–8]. Experimentally, the train of periodic pulses is applied for seconds to minutes which implies up to 10^{10} pulses because the generic repetition period T_R of the pulses is of the order of

P. Schering (✉) · P. W. Scherer · G. S. Uhrig
Condensed Matter Theory, TU Dortmund University, 44221 Dortmund, Germany
e-mail: philipp.schering@tu-dortmund.de

G. S. Uhrig
e-mail: goetz.uhrig@tu-dortmund.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_8

10 ns. Since the period of the electronic Larmor precession is of the order of 10 ps, the theoretical simulations have to cover 13 orders of magnitude in time; this is a tremendous computational challenge even for high performance computing (HPC).

It appears that the application of periodic pulses with repetition period T_R synchronizes the Larmor precessions of the spins in sub-ensembles of QDs, eventually leading to constructive interference of the Larmor precessions before each pulse accompanied by a revival of the spin polarization. The Overhauser field, i.e., the magnetic field applied by all the nuclear spins together via the hyperfine interaction on the electronic spin, changes slightly such that it compensates the fluctuations in the g factor from dot to dot which otherwise would lead to fast dephasing of the electronic Larmor precessions of different dots. This phenomenon is called nuclei-induced frequency focusing [8] and it is the effect which we study by quantitative simulations possible thanks to high performance computing. Thereby, we pave the way for future experiments exploiting the electronic spin in QDs as a quantum resource. The long-term goal is to generate coherent states including the nuclear spin degrees of freedom in single and multiple QDs by suitable pulse protocols, thereby lifting coherent control to another level.

Recent experiments on spin inertia and polarization recovery in QDs [9] revealed results which could not be fully explained by the analytic theoretical model [10]. By utilizing the high performance facilities of the HLRS, we can perform improved simulations of these experiments [11], which help us to gain a better understanding of the underlying physics. Importantly, we find the emergence of resonant spin amplification in the so called Faraday geometry, i.e., when a longitudinal magnetic field is applied to the QDs. This new effect can be revealed experimentally using optimized pulse protocols. Preliminary experimental results confirm its existence.

2 Nuclei-Induced Frequency Focusing in Quantum Dots

We investigate the spin dynamics of an inhomogeneous ensemble of GaAs QDs in a transverse magnetic field (Voigt geometry). Each QD is singly charged by a localized electron, whose spin couples to the surrounding nuclear spins via the hyperfine interaction. This electronic spin is excited optically by trains of resonant laser pulses with repetition period $T_R = 13.2$ ns, generating negatively charged trion states (transition energy $\sim$1.4 eV [6]). The trion eventually recombines, inducing some electronic spin polarization into the system due to the spin dynamics in combination with the selection rules. This polarization dephases on a timescale of nanoseconds due to the random nuclear spin bath, whose collective hyperfine interaction acts as Overhauser field on the electronic spin, and due to the spread of the electronic g factors, which differ slightly from dot to dot. Experimentally, the spin polarization can be probed using weak linearly polarized pulses by measuring the Faraday rotation or ellipticity [12, 13].

Upon application of long pulse trains, the spin dynamics reacts in such a way that a revival of the spin polarization emerges before the arrival of each next pulse. This effect is known as spin mode locking [7]. It can be enhanced by the fascinating phenomenon of nuclei-induced frequency focusing [8], which is one of the central subjects of the present report. The periodic pumping of the electronic spin indirectly drives the Overhauser field such that the effective Larmor frequency of the electronic spin in each QD complies with a certain resonance condition. Generically, this leads to an enhancement of the mode locked revival amplitude [8], but the dependence of this amplitude on the magnetic field strength is complex [14–16].

Here, we report on the recent progress in the simulation of this type of experiment. For a more detailed description of the simulations, the results, and the physics, we refer the reader to Ref. [17]. In order to render the simulation possible, progress in several key areas was required. First, we enhanced the semiclassical model describing the physical system and the optical generation of spin polarization via the excitation of a trion. The subsequent application of an efficient algorithm to the equations of motion, reducing the dimension of the system, is mandatory to deal with large bath sizes [18, 19]. In order to be able to make statements for the relevant number of total pulses, i.e., after which the system is in a nonequilibrium stationary state (NESS) as in the experiment, a very efficient and highly parallel implementation is required to solve the equations of motion. Even then it is not possible to deal directly with realistic bath sizes of up to 10^6 nuclear spins. We overcome this obstacle using established scaling arguments by which we can extrapolate to an infinite bath size.

2.1 *Hyperfine Interaction of an Electronic Spin with a Nuclear Spin Bath*

The dominant interaction in a GaAs QD singly charged by electrons is the Fermi contact hyperfine interaction [3]. In each QD, the quantum mechanical behavior of the spins is governed by the Hamiltonian

$$\hat{\mathcal{H}}_{\mathrm{hf}} = \sum_{k=1}^{N} A_k \hat{\boldsymbol{S}} \cdot \hat{\boldsymbol{I}}_k = \hat{\boldsymbol{S}} \cdot \hat{\boldsymbol{B}}_{\mathrm{ov}}, \tag{1}$$

in which the nuclear spins $\hat{\boldsymbol{I}}_k$ weighted by their hyperfine coupling constant A_k form the so called Overhauser field

$$\hat{\boldsymbol{B}}_{\mathrm{ov}} = \sum_{k=1}^{N} A_k \hat{\boldsymbol{I}}_k, \tag{2}$$

which couples to the central electronic spin $\hat{\boldsymbol{S}}$.

Solving the full quantum model is extremely restricted in the number of nuclear spins N due to the exponentially growing Hilbert space. We resort to an established semiclassical description of the problem, where the spins are considered as classical vectors with random initial conditions [20–22]. In this approach, the spin dynamics are governed by the classical equations of motion ($\hbar$ is set to unity) [17]

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{S} = (\boldsymbol{B}_{\mathrm{ov}} + g_{\mathrm{e}}\mu_{\mathrm{B}}B_{\mathrm{ext}}\boldsymbol{e}_x) \times \boldsymbol{S} + \frac{1}{\tau_0}J^z\boldsymbol{e}_z, \tag{3a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{J} = \chi\left(B^z_{\mathrm{ov}}\boldsymbol{e}_z + \frac{1}{\lambda}\boldsymbol{B}^{\perp}_{\mathrm{ov}}\right) \times \boldsymbol{J} + g_{\mathrm{h}}\mu_{\mathrm{B}}B_{\mathrm{ext}}\boldsymbol{e}_x \times \boldsymbol{J} - \frac{1}{\tau_0}\boldsymbol{J}, \tag{3b}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{I}_k = \left[A_k\boldsymbol{S} + \chi A_k\left(J^z\boldsymbol{e}_z + \frac{1}{\lambda}\boldsymbol{J}^{\perp}\right) + g_{\mathrm{n}}\mu_{\mathrm{n}}B_{\mathrm{ext}}\boldsymbol{e}_x\right] \times \boldsymbol{I}_k, \tag{3c}$$

where $\boldsymbol{B}^{\perp} := B^x\boldsymbol{e}_x + B^y\boldsymbol{e}_y$ and $\boldsymbol{J}^{\perp} := J^x\boldsymbol{e}_x + J^y\boldsymbol{e}_y$. These equations essentially describe a precession of the classical spins. The electronic spin $\boldsymbol{S}$ precesses around the effective magnetic field $\boldsymbol{B}_{\mathrm{eff}} := (\boldsymbol{B}_{\mathrm{ov}} + g_{\mathrm{e}}\mu_{\mathrm{B}}B_{\mathrm{ext}}\boldsymbol{e}_x)/(g_{\mathrm{e}}\mu_{\mathrm{B}})$, where B_{ext} is the strength of the external transverse magnetic field, $g_{\mathrm{e}} = 0.555$ [8] the g factor of the electronic spin and μ_{B} the Bohr magneton. The same holds for the trion pseudospin $\boldsymbol{J}$, but its hyperfine interaction is weaker by a factor $\chi \approx 0.2$ and also anisotropic ($\lambda = 5$) [9]. Moreover, the trion decays radiatively on the timescale $\tau_0 = 400\,\mathrm{ps}$ [6, 7]. According to the selection rules, a recombination of the trion component J^z and the ground state $\boldsymbol{S}$ takes place. Spin polarization in the ground state is generated when this recombination does not occur with the exactly same spin quantum number, for instance in an applied transverse magnetic field inducing Larmor precessions with different frequencies for $\boldsymbol{S}$ and $\boldsymbol{J}$. The nuclear spins $\boldsymbol{I}_k$ also precess around the so called Knight field plus the external magnetic field. Due to the larger masses of the nuclei, their gyromagnetic ratio $g_{\mathrm{n}}\mu_{\mathrm{n}}$ is smaller than $g_{\mathrm{e}}\mu_{\mathrm{B}}$ by three orders of magnitude. Nevertheless, the nuclear Zeeman term has a crucial impact on the nonequilibrium physics since the energy scale of A_k and $g_{\mathrm{n}}\mu_{\mathrm{n}}B_{\mathrm{ext}}$ can be of similar magnitude for large magnetic fields. The hyperfine couplings are parameterized according to

$$A_k \propto \exp(-k\gamma), \qquad k \in \{1, \ldots, N\}, \tag{4}$$

which is a realistic choice for flat two-dimensional QDs [18, 23, 24].

The ordinary differential equation system has the dimension $3N + 6$, where N is the total number of bath spins. Note that N is practically infinite in any solid state system. The number N_{eff} of effectively coupled spins within the localization volume of the electronic spin is much smaller, but still very large. For realistic bath sizes of at least $N = 10^4$, the numerical simulation of the desired properties is unfeasible even on a HPC system. We resort to the efficient approach established in Ref. [18], where sums of bath spins define auxiliary vectors. It is sufficient to track N_{tr} of these auxiliary vectors in a simulation. This reduces the dimension to $3N_{\mathrm{tr}} + 6$, where $N_{\mathrm{tr}} = \mathcal{O}(75)$ is a discretization parameter, while allowing us to treat an infinite spin bath $N \to \infty$ with an effective number $N_{\mathrm{eff}} \approx 2/\gamma$ of sizeably coupled nuclear spins.

The complexity of the equations does not increase. For details, we refer the interested reader to Ref. [18], where this approach is established, and to Refs. [17, 19], where it is applied to study nonequilibrium spin dynamics in quantum dots.

Due to the large number of nuclear spins, the Overhauser field and the auxiliary fields essentially behave like classical fields. Furthermore, the large number of contributing spins meets the precondition to use the central limit theorem to conclude that the Overhauser field and the auxiliary fields are initially normal distributed so that the initial auxiliary vectors can be sampled from normal distributions. The corresponding mean values and variances are chosen such that they mimic quantum mechanical properties [18]. The theoretical foundation for this approach is the truncated Wigner approximation [25]. First order quantum fluctuations are taken into account through the correct sampling of the initial conditions for the classical equations of motion.

Moreover, ensembles of QDs are not homogeneous. This leads to a slight spread of the g factors of the electronic spin and the trion pseudospin. We account for this spread by sampling the g factors from a normal distribution around their mean values with appropriate variances $(\Delta g)^2$.

The trion is excited by resonant circularly-polarized laser pulses; we consider so called π pulses with helicity σ^- here. They have a typical duration of 1.5 ps [6–8], which is one order of magnitude smaller than the Larmor period at a magnetic field of $B_{\text{ext}} = 9\,\text{T}$. Hence, we can describe the action of a single pulse as an instantaneous mapping of the spin components before ($\boldsymbol{S}_\text{b}$, $\boldsymbol{J}_\text{b}$) and after ($\boldsymbol{S}_\text{a}$, $\boldsymbol{J}_\text{a}$) the pulse [12, 15]

$$S_\text{a}^z = \frac{1}{4} + \frac{1}{2} S_\text{b}^z, \qquad S_\text{a}^x = S_\text{a}^y = 0, \tag{5a}$$

$$J_\text{a}^z = S_\text{b}^z - S_\text{a}^z, \qquad J_\text{a}^x = J_\text{a}^y = 0. \tag{5b}$$

In order to mimic the quantum properties of the spins, we must consider each pulse as a quantum mechanical measurement. According to the uncertainty principle, the semiclassical description of the pulses becomes nondeterministic. We model this uncertainty for $\boldsymbol{S}$ by normal distributions with mean values given by Eq. (5a) and appropriate variance such that quantum mechanical property $\langle(\hat{S}^\alpha)^2\rangle = 1/4$, $\alpha \in \{x, y, z\}$, holds, see Ref. [17] for details. The pulse relations for the trion pseudospin (5b) remain unchanged.

2.2 *Spin Dynamics, Spin Mode Locking, and Nuclei-Induced Frequency Focusing*

The time evolution of the spin dynamics is given by the ensemble average over $M = 4800$ independent classical trajectories with random initial conditions. For this purpose, we use $2400 - 4800$ CPU cores on Hazel Hen, parallelized using pure MPI, i.e., we calculate 1 or 2 trajectories per core.

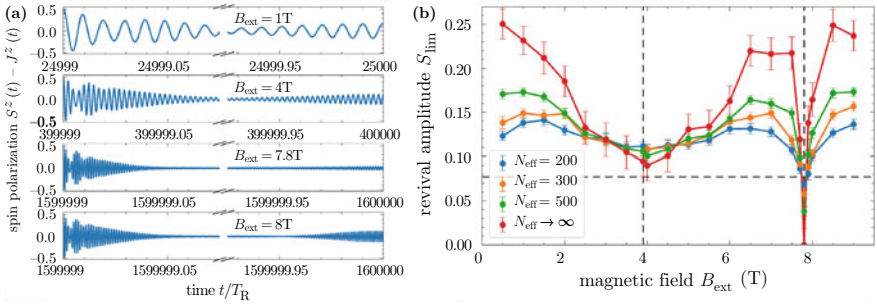


Fig. 1 (a) Spin dynamics after a long train of periodic pulses with repetition period T_R for various magnetic fields B_{ext} for an effective bath of $N_{eff} = 500$ nuclear spins. The oscillating spin polarization initially dephases, but a revival appears due to SML in combination with NIFF before the arrival of the next pulse. The amplitude of the revival signal does not change significantly anymore upon application of further pulses, i.e., the system is in a NESS. (b) Limiting values of the revival amplitude S_{lim} as a function of the magnetic field B_{ext} for various effective bath sizes N_{eff}. The horizontal dashed line represents the revival amplitude which emerges even without NIFF. The vertical dashed lines represent values of B_{ext} fulfilling the nuclear resonance condition (6). Further parameters: $g_h = 0.66$, $\Delta g_e = 0.005$, $\Delta g_h = 0.016$. Similar figures are shown in Ref. [17], but for different parameters.

Examples for the spin dynamics for different magnetic fields after a long train of periodic pulses are shown in Fig. 1a. The initially created spin polarization precesses around the transverse magnetic field while dephasing occurs on a timescale of 1 ns due to the fluctuations of the Overhauser field and due to the g factor spread. The dephasing is faster for larger magnetic fields since a g factor spread Δg implies a term proportional to $\Delta g B_{ext}$ in the dephasing rate [6]. Modulations of the signal are discernible in the time-dependence of the combined quantity $S^z - J^z$, which is proportional to the signal measured in experiments [12]. The modulations stem from the different Larmor frequencies of the electronic spin $\boldsymbol{S}$ and trion pseudospin $\boldsymbol{J}$.

After the initial dephasing, a revival appears before the arrival of each next pulse due to spin mode locking (SML). An initial buildup of this revival happens already within $\mathcal{O}(10)$ pulses. After a long train of pulses, the SML can change due to nuclei-induced frequency focusing (NIFF). Its amplitude depends on the strength of the external magnetic field due to its influence on NIFF, see Fig. 1. Note that for the data shown in Fig. 1a, the revival amplitude does not change noticeably anymore upon further application of pulses because the NESS is already reached. To be precise, a quasistationary state is reached which becomes apparent when studying the system stroboscopically at specific instants of the time interval between consecutive pulses, e.g., before the arrival of a pulse to study the revival amplitude.

This saturated revival amplitude S_{lim} as a function of the magnetic field shows an interesting nonmonotonic dependence on the magnetic field which is depicted in Fig. 1b. Two pronounced minima are visible at positions corresponding to the nuclear resonance condition [16, 17, 26]

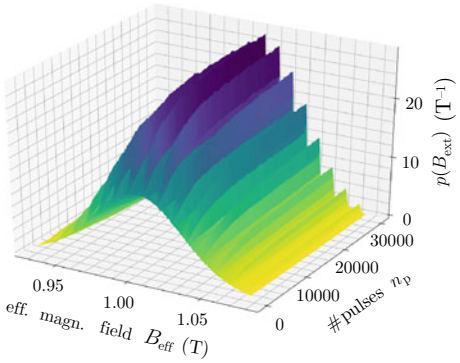


Fig. 2 Nuclei-induced frequency focusing: Probability distribution of the effective magnetic field $p(B_{\mathrm{eff}})$ as a function of the number of applied pulses n_{p}. Initially ($n_{\mathrm{p}} = 0$), the effective magnetic field follows a simple normal distribution. Equidistant peaks emerge in the distribution due to the application of periodic pulses. The peak positions correspond to the values of B_{eff} which fulfill the even resonance condition (7). Parameters: $B_{\mathrm{ext}} = 1\,\mathrm{T}$, $N_{\mathrm{eff}} = 500$, $g_{\mathrm{h}} = 0.66$, $\Delta g_{\mathrm{e}} = 0.005$, $\Delta g_{\mathrm{h}} = 0.016$.

$$g_{\mathrm{n}}\mu_{\mathrm{n}}B_{\mathrm{ext}}T_{\mathrm{R}} = \pi k, \qquad k \in \mathbb{Z}. \tag{6}$$

The values of B_{ext} fulfilling this condition are highlighted as vertical dashed lines in Fig. 1b. The condition describes the number of half-turn revolutions of the nuclear spins in the external magnetic field B_{ext} between consecutive pulses. Note that we consider only a single type of isotope for the nuclear spins here. Since generically there are several present in GaAs QDs, they can be viewed as an average isotope. This simplification is lifted later in Sect. 2.3. The horizontal dashed line represents the SML revival amplitude which emerges already without any NIFF. The comparison between the cases with and without NIFF is an important physical question addressed in Ref. [17].

The application of long trains of pulses leads to the emergence of a comblike structure in the probability distribution $p(B_{\mathrm{eff}})$ of the effective magnetic field. This frequency focusing in the nuclear spin bath is what causes the NIFF in the spin dynamics. The buildup of the comblike structure is illustrated in Fig. 2. The peak positions are found at the values of the effective magnetic field B_{eff} which fulfill the resonance condition

$$g_{\mathrm{e}}\mu_{\mathrm{B}}B_{\mathrm{eff}}T_{\mathrm{R}} = 2\pi k, \qquad k \in \mathbb{Z}. \tag{7}$$

This condition describes full-turn Larmor periods of the electronic spin between consecutive pulses. We refer to it as the *even* resonance condition because $2k$ is an even integer. In some cases, e.g., for $B_{\mathrm{ext}} \approx 7.8\,\mathrm{T}$, we find peaks at values of the effective magnetic field which fulfill the so called *odd* resonance condition

$$g_e \mu_B B_{\text{eff}} T_R = (2k+1)\pi, \qquad k \in \mathbb{Z}. \tag{8}$$

It describes half-turn Larmor periods between consecutive pulses and leads to a reduced revival amplitude in comparison to the case without nuclei-induced frequency focusing, see Fig. 1 for $B_{\text{ext}} = 7.8\,\text{T}$. Further insight into the importance of the two different resonance conditions is presented in Refs. [15–17, 26].

Furthermore, we find that the probability distribution of the effective magnetic field can shift as a whole; this effect is known as dynamic nuclear polarization (DNP) [17]. This polarization can be larger than the typical fluctuations of the Overhauser field, leading to a certain increase of the coherence time. Reaching the corresponding NESS in the simulations requires two orders of magnitude more pulses than reaching steady values of the revival amplitude. This renders the reliable simulation of the DNP unfeasible for magnetic fields much larger than 2 T and imposes an important challenge for further improvements of the employed algorithms.

Generally, simulating large effective bath sizes and large magnetic fields is a tremendous challenge. The number of pulses required to reach the steady values of the revival amplitude scales with B_{ext}^2 and linearly with N_{eff} [17, 19]. Moreover, in order to track the fast Larmor precession the integration step size decreases approximately with B_{ext}^{-1} so that we are facing a cubic scaling in the computational complexity for larger fields. Developing performant approaches to mitigate this problem is part of our current research. A first approach of this kind, which is already applied in the present simulations, is briefly discussed in Sect. 4.

2.3 Role of the Nuclear Spin Bath Composition

Real QD ensembles studied in experiments do not consist of a single isotope but of many, e.g., they are GaAs or InGaAs QDs. This can be accounted for in the equations of motion (3), but it increases the dimension of the ordinary differential equation system and becomes intractable when studying the exponential parameterization (4) of the hyperfine couplings. A common simplification is the so called 'box model', where all couplings are chosen to be equal [27]. Then, the full dynamics of the Overhauser field can be described by a single equation of motion for the subfield made up of each different isotope in the system. For brevity, we omit the precise equations here.

We apply a simpler pulse model than in the previous Sect. 2.1 based on Ref. [19], where we completely neglect the excitation of the trion state and instead describe the pulse action by the simple relation

$$S_a^z = \frac{1}{2}, \qquad S_a^x = X, \qquad S_a^y = Y, \tag{9}$$

with X and Y being random numbers sampled from a normal distribution around zero with variance $1/4$, i.e., we still consider each pulse as a quantum mechanical

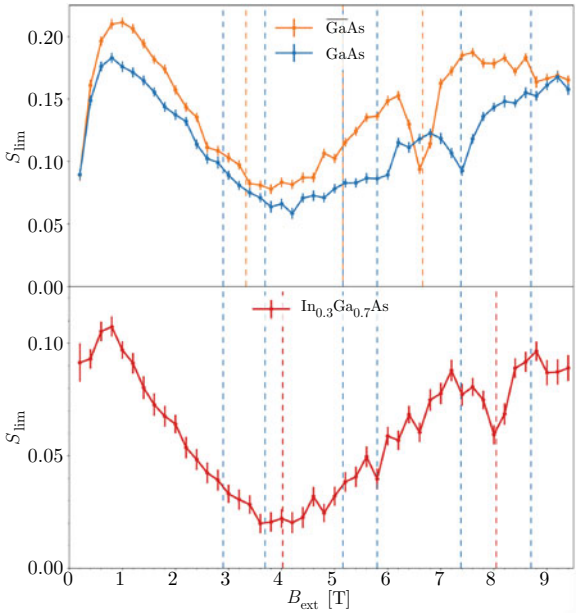


Fig. 3 Saturated revival amplitude S_{lim} as a function of the magnetic field B_{ext} for different compositions of the nuclear spin bath ($N = 100$). In the upper panel the dephasing time is $T_{\text{n}}^* = \sqrt{2}\,\text{ns}$, in the lower panel it is $T_{\text{n}}^* = 1\,\text{ns}$. The horizontal dashed lines indicate the values of B_{ext} which fulfill the nuclear resonance condition (6) for the various isotopes.

measurement. We point out that this pulse model does not show SML without NIFF, i.e., for a small number of pulses no revival amplitude prior to the next pulse occurs. Only long trains of pulses leading to NIFF engender a revival amplitude. This is a fundamental difference to the more elaborate pulse model used in Sect. 2.1 and the main downside of neglecting the generation of spin polarization by means of an intermediate trion state. Still, the nondeterministic description of the pulse is essential to mimic the quantum mechanical behavior [16, 17, 19]. In contrast to the previous Sect. 2.1, we omit the g factor spread here, but it barely affects the NIFF behavior [17].

When studying nuclear spin baths consisting of various isotopes, we expect additional nuclear resonance conditions (6) to play a role because each isotope has a different gyromagnetic ratio $g_{\text{n}}\mu_{\text{n}}$. Figure 3 shows the magnetic field dependence of the revival amplitude for three different compositions of the nuclear spin bath. In the upper panel, we compare the result for a GaAs QD with the case where a GaAs QD with an average Ga isotope is considered ($\overline{\text{GaAs}}$). In the lower panel, we show the result for an $\text{In}_{0.3}\text{Ga}_{0.7}\text{As}$ QD. A quantitative comparison of both panels is not possible yet because the dephasing times T_{n}^*, which determine the variance of the random Overhauser field, are chosen differently. The main conclusion is that not all nuclear resonance conditions play a major role as we still mainly find a broad

minimum around $B_{\text{ext}} = 4\,\text{T}$ and a sharper minimum at larger magnetic fields. Some additional structure is visible around $B_{\text{ext}} = 5.8\,\text{T}$, but further more accurate calculations are required for a better resolution. Due to the increased complexity of the physical situation, the degree of NIFF is reduced when the number of different isotopes is increased. Furthermore, indium dominates the behavior of the system even for very small concentrations because it has spin $9/2$ whereas the other isotopes have spin $3/2$, and its hyperfine coupling constant is also larger [28].

3 Spin Inertia and Polarization Recovery in Quantum Dots

In the previous sections, we considered the application of a transverse magnetic field (Voigt geometry). In a different class of pump-probe experiments on QD ensembles, a longitudinal magnetic field (Faraday geometry) is applied. Here, we consider QDs charged by either electrons or holes. Typical experiments study the occuring spin inertia and polarization recovery effects [9–11, 29]. Periodic circularly-polarized laser pulses are applied with repetition period $T_{\text{R}} = 13.2\,\text{ns}$, but their helicity is modulated between σ^+ and σ^- with frequency f_{m}. Studying the spin polarization as a function of this modulation frequency shows the so called spin inertia effect: when the modulation frequency is increased, the spin polarization decreases. This can be understood as an inertia of the spin which prevents it from following the switching of the pulse helicity arbitrarily quickly. This effect enables the measurement of slow relaxation times $\mathcal{O}(\mu\text{s})$ of the system. The polarization recovery effect consists of the increase of the spin polarization upon an increase of the longitudinal magnetic field. Hence, the polarization recovery curve is the graph of the spin polarization as a function of this magnetic field. We report on the influence of the pumping strength on these experiments; for details see Ref. [11].

The simulation of this setup is less demanding than the one of Sect. 2 because the dynamics of the Overhauser field plays a minor role so that it can be considered as frozen [27], i.e., as static, but still random according to a normal distribution to account for its quantum mechanical fluctuations. Then, the spin dynamics of the localized charge carrier in the ground state of each QD is described by

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{S} = \left(\boldsymbol{\Omega}_{\text{N,g}} + \boldsymbol{\Omega}_{\text{L,g}}\right) \times \boldsymbol{S} - \frac{\boldsymbol{S}}{\tau_{\text{s,g}}} + \frac{J^z}{\tau_0}\boldsymbol{e}_z, \tag{10}$$

where $\boldsymbol{\Omega}_{\text{N,g}}$ is the frequency of the spin precession caused by the Overhauser field, $\boldsymbol{\Omega}_{\text{L,g}} = \Omega_{\text{L,g}}\boldsymbol{e}_z = g_{\text{g}}\mu_{\text{B}}B_{\text{ext}}\boldsymbol{e}_z$ is the Larmor frequency, with g_{g} being the effective longitudinal g factor of the ground state, and $B_{\text{ext}}\boldsymbol{e}_z$ the external longitudinal magnetic field. Furthermore, the phenomenological term $-\boldsymbol{S}/\tau_{\text{s,g}}$ describes the spin relaxation unrelated to the hyperfine interaction with the nuclear spins in the QD. The dynamics of the trion pseudospin between the pump pulses is described similarly to Eq. (10) by the equation of motion

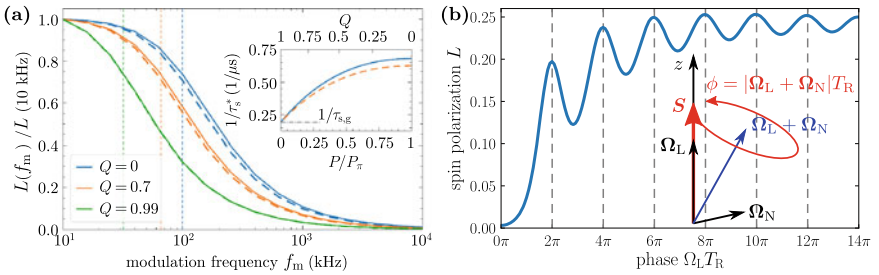


Fig. 4 (a) Spin inertia effect for localized holes in quantum dots: spin polarization L as a function of the modulation frequency f_m for a magnetic field of $B_{ext} = 300$ mT and various pumping efficiencies Q. The vertical dashed lines represent the typical cut-off frequencies $1/(2\pi\tau_s^*)$. The inset shows the inverse of the effective spin relaxation time τ_s^* as a function of the pump power P and of the pump efficiency Q. The dashed curves are calculated analytically for the limit $B_{ext} \to \infty$. (b) Illustration of resonant spin amplification in Faraday geometry: the spin polarization increases as a function of the phase $\Omega_L T_R \equiv \Omega_{L,g} T_R \propto B_{ext}$ with periodic oscillations at positions fulfilling the resonance condition (12). Both figures are taken from Ref. [11], lincensed under CC BY 4.0 (https://creativecommons.org/licenses/by/4.0/), with slight modifications to the layout.

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{J} = \left(\boldsymbol{\Omega}_{N,t} + \boldsymbol{\Omega}_{L,t}\right) \times \boldsymbol{J} - \frac{\boldsymbol{J}}{\tau_{s,t}} - \frac{\boldsymbol{J}}{\tau_0}. \tag{11}$$

The Overhauser field again follows a normal distribution, but it can be anisotropic for hole spins [30], and the spin polarization of the QD ensemble is calculated by averaging over all trajectories stemming from random initial conditions. Details of the simulation for QDs charged by either electrons or holes can be found in the original publication [11]. The computations are not as expensive as in the previous sections because we only need to study magnetic fields up to 300 mT and overall a smaller number of pulses. The periodic laser pulses are described by a similar but more general relation than given by Eq. (5), which includes possible rotations of the transverse spin components and also allows us to consider various pump pulse efficiencies [11].

Our main results from Ref. [11] are the following. In accordance with experiments [9], we find that applying a larger pump power leads to a decrease of the effective spin relaxation time in the system. This behavior is visualized in the inset of Fig. 4a for the case of localized holes in the QDs. The inverse of the effective spin relaxation time is plotted as a function of the pump power P and of the pumping efficiency Q for a magnetic field of $B_{ext} = 300$ mT. The best pumping efficiency is achieved for $Q = 0$, which describes so called π pulses reached at the pump power P_π. Weak pulses are described by the limit $Q \to 1$. A linear extrapolation to zero pump power yields the equilibrium spin relaxation time $\tau_{s,g}$. The main panel of Fig. 4a shows the spin inertia effect for various pumping efficiencies Q. Upon increasing the modulation frequency f_m, the spin polarization decreases. Analytical results for the limit $B_{ext} \to \infty$ (dashed curves) support our findings. The effective spin relaxation time can be extracted from the dependence of the spin polarization on the modulation frequency.

Furthermore, we analyze the role of the saturation of spin polarization in polarization recovery measurements. We find that approaching the saturation limit of the spin polarization leads to a broadening of the typical V-like shape of the polarization recovery curves (the curves are symmetric with respect to the magnetic field B_{ext}), similar to what is observed experimentally [9].

Most importantly, we find the emergence of resonant spin amplification in Faraday geometry [11]. It is a well established effect in Voigt geometry [31, 32] and also known for a tilted magnetic field [33], but it is to our knowledge not yet discovered in a pure longitudinal field configuration. It can emerge purely due to the transverse fluctuations of the Overhauser field in the QDs under certain conditions and it can be exploited to measure the longitudinal g factor g_{g} of the localized charge carriers in the QDs. We stress, however, that the longitudinal fluctuations of the Overhauser field may not be too large since the effect is smeared out otherwise. A typical polarization recovery curve illustrating this effect is shown in Fig. 4b. The spin polarization increases for larger magnetic fields B_{ext}, which is proportional to the phase $\Omega_{\mathrm{L}} T_{\mathrm{R}} \equiv \Omega_{\mathrm{L,g}} T_{\mathrm{R}} \propto B_{\mathrm{ext}}$ displayed on the abscissa. Periodic oscillations are found whenever the resonance condition

$$\Omega_{\mathrm{L,g}} T_{\mathrm{R}} = 2\pi k, \qquad k \in \mathbb{Z}, \tag{12}$$

is fulfilled. This effect is caused by the slight tilt of the effective magnetic field $\mathbf{\Omega}_{\mathrm{L,g}} + \mathbf{\Omega}_{\mathrm{N,g}}$ from the z axis due to the transverse components of the Overhauser field $\mathbf{\Omega}_{\mathrm{N,g}} \equiv \mathbf{\Omega}_{\mathrm{N}}$ in each QD, see the sketch in Fig. 4b for a graphical illustration. We estimate that resonant spin amplification in Faraday geometry can be observed under the condition $\omega_{\mathrm{n,g}} \lesssim \sqrt{2}\pi / T_{\mathrm{R}}$, where $\omega_{\mathrm{n,g}}$ is the typical fluctuation strength of the Overhauser field, when sufficiently strong pump pulses are applied. This condition implies that reducing the pulse repetition period T_{R} can help to reveal this new effect. In the experiments of Ref. [9], the effect is not observed because the condition is not fulfilled and pump pulses of low power are used.

Note the similarity to the resonance conditions discussed in Sect. 2. The resonance conditions are central to understanding the nonequilibrium spin dynamics in QDs subjected to trains of periodic pulses whenever an external magnetic field is applied. Preliminary experimental results indicate that the effect depicted in Fig. 4b can indeed be measured.

4 Efficient Simulations

Solving ordinary differential equations (ODEs) is straightforward, especially if no stability problems occur as in our case. We apply the Dormand-Prince method as ODE solver, which is an adaptive fifth-order Runge-Kutta algorithm, using the implementation provided in Ref. [34]. The simulation of realistic experimental setups is extremely challenging because the relevant time scales govern several orders of magnitude. For this reason, on the one hand, the integration error must be small enough

so that errors do not add up significantly after up to millions of pulses. On the other hand, one cannot aim at reducing it too much by very small time steps because this would spoil the performance.

Reaching a NESS as required in Sect. 2 for large magnetic fields to study the NIFF behavior is extremely demanding because the computational complexity scales with B_{ext}^3. Simulating large spin baths is an additional challenge due to a linear scaling with N_{eff} of the numbers of pulses required to reach the quasistationary states. In Ref. [11], we also established scaling laws which allow us to extrapolate to infinite bath sizes $N_{\mathrm{eff}} \to \infty$. The direct simulation of such bath sizes is not possible even by high performance computing.

4.1 Spectral Density and Rotating Frame Approach

As discussed in Sect. 2.1, the application of an efficient approach [18] to the ODE system (3) reduces its dimension from $3N + 6$ to $3N_{\mathrm{tr}} + 6$, with the truncation parameter $N_{\mathrm{tr}} \ll N$, by replacing sums of bath spins by auxiliary vectors. In a typical scenario, $N_{\mathrm{tr}} = 75$ for an effective bath size of $N_{\mathrm{eff}} = 200$ with $N \to \infty$ is used. If we solved the normal ODE system (3) for $N = 200$ nuclear spins, the dimension would be about 2.5 times larger, eventually requiring the usage of the slower L2 and L3 caches for large spin baths.

Further performance improvements are gained by solving the ODE system (3) in a rotated frame. The electronic spin $\boldsymbol{S}$ and the trion pseudospin $\boldsymbol{J}$ mainly precesses around the large transverse external magnetic field, and this precession is by far the fastest frequency in the system. Thus, it determines the integration step size in a linear manner, i.e., it is proportional to B_{ext}^{-1}. However, this precession motion can be easily described analytically. The application of such an ansatz to the ODE system yields modified equations of motion, and the integration of the ODE system is about three times faster while maintaining the same numerical accuracy. Unfortunately, the step size still decreases linearly when increasing the magnetic field strength: the nuclear spins also show the fast oscillations due to their coupling to the precessing electronic spin, which need to be resolved numerically.

Establishing even more efficient approaches is part of our current research. One particular goal is an approach in which the integration step size is not determined linearly by the magnetic field strength since this would render the simulation of large magnetic fields much more efficient.

4.2 Single Thread Performance

Vectorization is the key to achieve good single thread performance. The applied Runge-Kutta algorithm is easily vectorized by any modern compiler. For the ODE system (3), a long loop must iterate over all nuclear spins, but the structure of their

equations of motion is identical. By using a structure of arrays for the data layout, good vectorization is accomplished. Due to the rather small dimension of the ODE system (about 230 equations), the calculations are operating on the L1 cache. We measure a double performance of 11.2 GFLOPS per core on a dual socket Intel Xeon Broadwell E5-2680 v4 (2x14 cores, 2.4 GHz) without hyperthreading, with a vectorization ratio of 91%.

Vectorization is not straight forward when we apply the box model to the nuclear spins as in Sect. 2.3 because there is no long loop which iterates over the nuclear spins. The typical dimension of a single ODE system of 9 to 18 is small. But we have to solve the ODE system for $\mathcal{O}(10^4)$ independent initial conditions and thus, we can group them together and solve their equations of motion simultaneously, i.e., we consider the group as a single combined ODE system. Again, using a structure of arrays for the data layout, good vectorization of the code is accomplished. The maximal group size is limited by the size of the L1 cache; the precise group size is chosen to optimize the performance.

The same principle is applied when solving the ODE sytem of Sect. 3 with a dimension of 6. When building groups of 64 independent random initial configurations, we measure a double precision performance of 9.7 GFLOPS per core on a dual socket Intel Xeon Broadwell E5-2680 v4 (2x14 cores, 2.4 GHz) without hyperthreading, with a vectorization ratio of 99%. Without the grouping procedure, the performance decreases to mere 2.0 GFLOPS per core with a vectorization ratio of only 13%.

4.3 Parallelization

The semiclassical simulation of the spin dynamics is easily parallelizable using pure MPI since the calculation of a single trajectory does not depend on the other ones. Minor communication takes place only at the end of the simulation where ensemble averages are calculated and the Overhauser fields are stored for the subsequent statistical analysis. The influence of the minor I/O on the performance is negligible.

Figure 5 shows a typical scaling behavior of our simulation code used for the simulations presented in Sect. 2.2. Very good scaling is achieved up to 12288 cores when calculating $M = 24576$ trajectories. In this extreme scenario, two trajectories are calculated per core. Due to the adaptive integration of the ODE system, some integrations finish sooner than others, which leads to a slightly reduced parallel efficiency when only few trajectories are calculated per core. In practice, we typically use 2400 cores to calculate 4800 independent trajectories, i.e., we calculate two trajectories per core. For large magnetic fields combined with large bath sizes, it is required to use 4800 cores for the same number of trajectories such that the simulation finishes within 24 h.

In cases where a deterministic pulse model is applied, a slightly better load balance can be achieved by running a short benchmark of about ten pulses for the calculation of each trajectory in the beginning, i.e., before the actual simulation. Then, slow

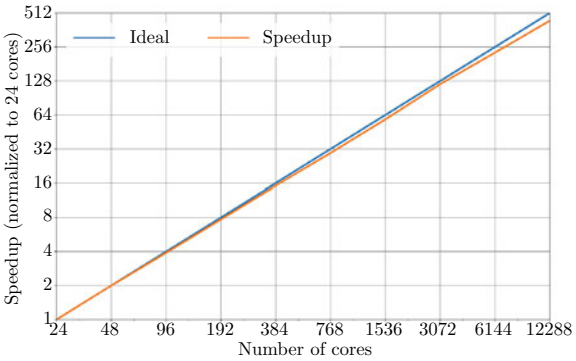


Fig. 5 Logarithmic plot of the scaling behavior of our code used in Sect. 2.2 on Hazel Hen (2x Intel Xeon Haswell E5-2680 v3 (2.5 GHz, 12 cores) per node) at HLRS. The relative speedup normalized to 24 cores is shown and compared to the ideal case. In this benchmark, a total of $M = 24576$ trajectories are calculated.

and fast calculations can be grouped together to improve the load balance. This procedure does not work reliably when the pulses are nondeterministic because the short benchmarks are not representative anymore for the full integration. Generally, when more than one trajectory is calculated per core, the load balance improves automatically due to a self-averaging effect.

As demonstrated in Fig. 5, the number of total cores can be increased well beyond the 4800 cores typically used in our simulations if a better statistical accuracy, accomplished by a larger ensemble size M, is desired. The statistical errors of the ensemble averages are proportional to $1/\sqrt{M}$.

5 Conclusion

Our large-scale simulations of the nonequilibrium spin dynamics in quantum dots (QDs) subjected to trains of periodic laser pulses pave the way for a better understanding and description of related pump-probe experiments [6–9, 14, 16]. By combining sophisticated efficient approaches with the computation power of Hazel Hen provided by the HLRS, we are able to reach nonequilibrium (quasi)stationary states for the full range of magnetic fields studied in the experiments for large bath sizes. Further scaling relations are established which enable us to extrapolate to infinite bath sizes, which is the physical limit of interest.

Meanwhile, we explore the influence of the nuclear spin bath composition and find that for InGaAs QDs, indium plays the most important role due to its large spin 9/2 and its larger hyperfine coupling. The degree of nuclei-induced frequency focusing is reduced when more complex compositions are considered. The combination of this enhanced and more realistic description of the nuclear spin bath with our more elaborate model of Sect. 2.1 comprising the generation of spin polarization via trion excitation is an imminent part of our current research.

The direct numerical simulation of the spin inertia and polarization recovery experiments for arbitrary pumping strength yields a better understanding of the experi-

mental results presented in Ref. [9]. The existing analytic theoretical description [10] is only valid in the low pump power limit. We identify the influence of large pump powers leading to saturation effects in the measurements. Importantly, we find the emergence of resonant spin amplification in Faraday geometry. It allows for measuring the longitudinal g factor of the localized charge carriers in the QDs. Preliminary results from our experimental colleagues suggest that they can indeed detect the predicted effect. The quantitative description of their measurements requires the extension of the current model to account for the inhomogeneous broadening of the trion transition energy [12], which is present in any real QD ensemble. This extension adds another stochastic component to the simulation, leading to a further increase of the computational complexity which we will address with the help of high performance computing.

Acknowledgements We thank Carsten Nase and Dmitry S. Smirnov for helpful discussions and gratefully acknowledge the Gauss Centre for Supercomputing e.V. for supporting this project by providing computing time on the GCS Supercomputer Hazel Hen at Höchstleistungsrechenzentrum Stuttgart (HLRS). This study has been funded by the German Research Foundation (DFG) and the Russian Foundation for Basic Research (RFBR) in the International Collaborative Research Centre TRR 160 (Projects No. A4 and No. A7).

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. D. Loss, D.P. DiVincenzo, Phys. Rev. A **57**, 120 (1998). https://doi.org/10.1103/PhysRevA.57.120
2. M.A. Nielsen, I.L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2000)
3. B. Urbaszek et al., Rev. Mod. Phys. **85**, 79 (2013). https://doi.org/10.1103/RevModPhys.85.79
4. J. Schliemann, A. Khaetskii, D. Loss, J. Phys. Condens. Matter **15**, R1809 (2003). https://doi.org/10.1088/0953-8984/15/50/r01
5. R. Hanson, L.P. Kouwenhoven, J.R. Petta, S. Tarucha, L.M.K. Vandersypen, Rev. Mod. Phys. **79**, 1217 (2007). https://doi.org/10.1103/RevModPhys.79.1217
6. A. Greilich et al., Phys. Rev. Lett. **96**, 227401 (2006). https://doi.org/10.1103/PhysRevLett.96.227401
7. A. Greilich et al., Science **313**, 341 (2006). https://doi.org/10.1126/science.1128215
8. A. Greilich et al., Science **317**, 1896 (2007). https://doi.org/10.1126/science.1146850
9. E.A. Zhukov et al., Phys. Rev. B **98**, 121304(R) (2018). https://doi.org/10.1103/PhysRevB.98.121304
10. D.S. Smirnov et al., Phys. Rev. B **98**, 125306 (2018). https://doi.org/10.1103/PhysRevB.98.125306
11. P. Schering, G.S. Uhrig, D.S. Smirnov, Phys. Rev. Res. **1**, 033189033189 (2019). https://doi.org/10.1103/PhysRevResearch.1.033189
12. I.A. Yugova, M.M. Glazov, E.L. Ivchenko, A.L. Efros, Phys. Rev. B **80**, 104436 (2009). https://doi.org/10.1103/PhysRevB.80.104436
13. M.M. Glazov, Phys. Solid State **54**, 1 (2012). https://doi.org/10.1134/S1063783412010143

14. S. Varwig, A. Greilich, D.R. Yakovlev, M. Bayer, Phys. Status Solidi B **251**, 1892 (2014). https://doi.org/10.1002/pssb.201350235
15. N. Jäschke et al., Phys. Rev. B **96**, 205419 (2017). https://doi.org/10.1103/PhysRevB.96.205419
16. I. Kleinjohann et al., Phys. Rev. B **98**, 155318 (2018). https://doi.org/10.1103/PhysRevB.98.155318
17. P. Schering, P.W. Scherer, G.S. Uhrig, Phys. Rev. B **102**, 115301 (2020). https://doi.org/10.1103/PhysRevB.102.115301
18. B. Fauseweh, P. Schering, J. Hüdepohl, G.S. Uhrig, Phys. Rev. B **96**, 054415 (2017). https://doi.org/10.1103/PhysRevB.96.054415
19. P. Schering, J. Hüdepohl, G.S. Uhrig, B. Fauseweh, Phys. Rev. B **98**, 024305 (2018). https://doi.org/10.1103/PhysRevB.98.024305
20. G. Chen, D.L. Bergman, L. Balents, Phys. Rev. B **76**, 045312 (2007). https://doi.org/10.1103/PhysRevB.76.045312
21. D. Stanek, C. Raas, G.S. Uhrig, Phys. Rev. B **90**, 064301 (2014). https://doi.org/10.1103/PhysRevB.90.064301
22. L.P. Lindoy, T.P. Fay, D.E. Manolopoulos, J. Chem. Phys. **152**, 164107 (2020). https://doi.org/10.1063/5.0006411
23. W.A. Coish, D. Loss, Phys. Rev. B **70**, 195340 (2004). https://doi.org/10.1103/PhysRevB.70.195340
24. A. Faribault, D. Schuricht, Phys. Rev. Lett. **110**, 040405 (2013). https://doi.org/10.1103/PhysRevLett.110.040405
25. A. Polkovnikov, Ann. Phys. (NY) **325**, 1790 (2010). https://doi.org/10.1016/j.aop.2010.02.006
26. W. Beugeling, G.S. Uhrig, F.B. Anders, Phys. Rev. B **96**, 115303 (2017). https://doi.org/10.1103/PhysRevB.96.115303
27. I.A. Merkulov, A.L. Efros, M. Rosen, Phys. Rev. B **65**, 205309 (2002). https://doi.org/10.1103/PhysRevB.65.205309
28. W.A. Coish, J. Baugh, Phys. Status Solidi B **246**, 2203 (2009). https://doi.org/10.1002/pssb.200945229
29. F. Heisterkamp et al., Phys. Rev. B **91**, 235432 (2015). https://doi.org/10.1103/PhysRevB.91.235432
30. C. Testelin, F. Bernardot, B. Eble, M. Chamarro, Phys. Rev. B **79**, 195440 (2009). https://doi.org/10.1103/PhysRevB.79.195440
31. J.M. Kikkawa, D.D. Awschalom, Phys. Rev. Lett. **80**, 4313 (1998). https://doi.org/10.1103/PhysRevLett.80.4313
32. I.A. Yugova, M.M. Glazov, D.R. Yakovlev, A.A. Sokolova, M. Bayer, Phys. Rev. B **85**, 125304 (2012). https://doi.org/10.1103/PhysRevB.85.125304
33. E.A. Zhukov et al., Phys. Rev. B **86**, 245314 (2012). https://doi.org/10.1103/PhysRevB.86.245314
34. W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, *Numerical Recipes* The Art of Scientific Computing, 3rd edn. (Cambridge University Press, New York, 2007)

Tuning the Conductivity of Metallic Nanowires by Hydrogen Adsorption

Kris Holtgrewe, Ferdinand Ziese, Johannes Bilk, Mike N. Pionteck, Kevin Eberheim, Felix Bernhardt, Christof Dues, and Simone Sanna

Abstract Nearly one-dimensional metallic nanowires can be grown by Au deposition on semiconducting substrates. Although the wires are basically decoupled from the substrate, their conductivity can be tuned by modifications of the environment, such as the adsorption of foreign species. In this work, we present a detailed theoretical investigation of the metallicity modifications in a system consisting of self-assembled Au atomic wires on a stepped Si(557) surface upon hydrogenation. Our first-principles calculations within density functional theory (DFT) reveal chemically different favorable adsorption sites of similar energy for H. The H adsorption at all sites leads to strong modifications in the electronic structure, due to the adsorbate-to-substrate charge transfer. Interestingly, the whole electronic system, including the Au related electronic states, is modified by H adsorption, without direct interactions of H and the Au chains. Changes in band order as well as a band gap opening nicely explain recent plasmon spectroscopy experiments.

1 Introduction

Metallic nanowires grown on semiconducting surfaces are highly anisotropic objects characterized by peculiar physical properties that closely resemble those of one-dimensional (1D) objects [1, 2]. The semiconducting substrate stabilizes the atomic chains and affects to some extent the 1D properties. Thus, the coupling between the inherently 1D atomic chains and their 2D or 3D environment determines the system properties.

The interactions between the wires and their environment can be exploited to control the physics of the wires. Stepped substrates can be used, e.g., to separate the

K. Holtgrewe · C. Dues · S. Sanna (✉)
Institut für Theoretische Physik and Center for Materials Research (LaMa), Justus-Liebig-Universität Gießen, Heinrich-Buff-Ring 16, 35392 Gießen, Germany
e-mail: simone.sanna@theo.physik.uni-giessen.de

F. Ziese · J. Bilk · M. N. Pionteck · K. Eberheim · F. Bernhardt
Institut für Theoretische Physik, Justus-Liebig-Universität Gießen, Heinrich-Buff-Ring 16, 35392 Gießen, Germany

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_9

wires, as the terrace width determines the distance between parallel wires and thus their coupling. The adsorption of gases leads to the formation of chemical bonds and is known to affect the wire conductivity, as revealed by transport experiments and plasmon spectroscopy [3]. Concerning the latter, 1D wire plasmons with their intrinsic directionality of energy transport are particularly interesting for a multitude of applications [4–6]. The tunability of plasma frequencies makes atomic wires attractive for novel electric circuits on the atomic scale [7, 8]. Different studies, both theoretical and experimental, have been dedicated to this topic [9–12]. However, the precise knowledge of the effect of the environment on the electronic band structure is required to control the plasma frequency.

Among the semiconducting substrates employed for the wire growth, high-index Si(hhk) surfaces, are among the most investigated systems, as they provide tunable terrace size and local structural motifs, which can be exploited to control the wire coupling with higher dimensions [9, 12–15]. In this work, we use the Si(557)-Au stepped surface to reveal the subtle intertwinement between atomic structure and electronic states and demonstrate how to they can be manipulated by H adsorption. The latter can thus be employed to modify the system metallicity. In particular, we show how the H adsorption leads to modifications of the metallicity by affecting the bands between the Au states in the unoccupied part of the band structure. The modifications of the band structure furthermore explain the plasmon localization measured in recent experiments [16].

2 Methodology

Total energy density functional calculations have been performed within the generalized gradient approximation (GGA) [17], as implemented in the VASP simulation package [18, 19]. The package offers highly customizable parallelization schemes to exploit the full capability of massively parallel supercomputers such as the Cray XC40 system (Hazel Hen) at the HLRS. In order to separate the discussion about the computational setup and the discussion about the algorithm performances, we divide the present section in two parts.

2.1 Computational Details

Projector augmented wave (PAW) potentials [20, 21] with projectors up to $l = 3$ for Au (valence $5d^{10}6s^1$), $l = 2$ for Si (valence $3s^2 3p^2$), and $l = 1$ for H (valence $1s^1$) have been used for the calculations. k-point meshes of different sizes are employed, depending on the size of the investigated system. Corresponding information is given in Sect. 3. The electronic wave functions are expanded into plane waves up to a kinetic energy of 410 eV. We chose the PBE formulation [22] of the generalized gradient approximation for our calculations. PBE is a well-tempered balance between com-

putational efficiency, numerical accuracy, and reliability. Modified GGA functionals such as PBEsol have been recently proposed, which are specifically developed to improve the description of the lattice properties [23, 24]. As we mainly focus on the description of the electronic band structure, the choice of PBE as xc-functional seems to be appropriate. We model the system at the experimentally measured lattice constant, which strongly reduces possible uncertainties arising from the typical PBE-overestimation of about 1% of the lattice parameters. Our approach thus pragmatically combines a reliable description of the electronic properties (within the accuracy of DFT with (semi)local xc-functionals) and a reasonable description of the system geometry. The described approach is on the same footing of recent investigations of metallic Au nanowires on semiconducting substrates [3, 25].

2.2 Computational Performance

The parallelization of the computational algorithms is usually achieved via the Message Passing Interface (MPI). OpenMP or hybrid MPI/OpenMP parallelization is also possible. However, our tests show that best performance is reached with pure MPI parallelization (one MPI task for each physical core). The speed-up depends on the size of the problem. For the problem sizes in our project (several 100 atoms per unit cell), the wall time scales very well up to several 1000 MPI processes, as we will show in the following.

There are different levels of code parallelization, which can be controlled by the code input. The most important ones are the number of k-points treated in parallel (KPAR), and the subsequent number of bands treated in parallel (NPAR) in the matrix diagonalization process. They may be set independently from each other, however, they strongly influence the wall time needed for the job. Our experience shows that the performance is maximized when KPAR is a divisor of both the number of allocated computer nodes and the total number of k-points of the system. This can be explained by the fact that this choice distributes the workload over the nodes optimally, which prevents idle times. The fine tuning is done by setting NPAR, which also should be a divisor of the number of bands. The optimal parallelization setup depends on the host machine and its peculiar architecture. Hence, when running the code on a new machine, we have to do a performance test with varying number of MPI processes, KPAR and NPAR. From the test results, we can derive rules of thumb, which enables the guessing of optimal parameters for all other calculations.

Performance Test

In the following, we present the results of a performance test performed on the Cray XC40 (Hazel Hen).

As test problem, we set up a large supercell modelling $ErSi_2$ nanowires. It contains 500 atoms and its size is thus comparable to the supercells needed to model the systems investigated in this project. The performed task was the calculation of the electronic ground state (total energy calculation), which is the main undertaking

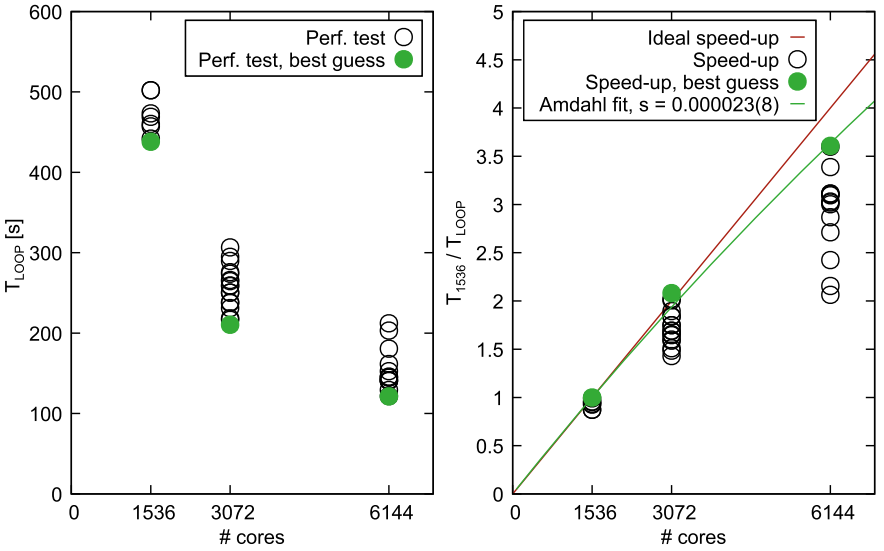


Fig. 1 Results of the performance test; left panel: test results for different parallelisation setups, right panel: speed-up of the best guess and Amdahl fits, asymptotic fit errors in parentheses

in the DFT framework. We chose the average wall time, which elapses for each electronic iteration T_{LOOP}, as target quantity. This choice makes the computational demand for every test run comparable.

We tested different parallelisation setups (varying number of nodes, KPAR, and NPAR). Thereby, we used complete nodes and one MPI task per physical core (24 MPI tasks per node). Figure 1, left panel shows the test results. One immediately sees that the wall times T_{LOOP} depend strongly on the choice of KPAR and NPAR (empty circles on vertical line at given number of cores). The setup with the minimum T_{LOOP} for given number of MPI tasks is called the "best guess".

As we are interested in the scalability of the code, we calculated the speed-up as the ratio between the loop wall time T_{1536} for 1536 employed cores and the loop wall time $T_{\text{LOOP}}(N)$ for N cores as we used 64 nodes at 24 cores at the minimum.

$$S_{1536}(N) = \frac{T_{1536}}{T_{\text{LOOP}}(N)} \tag{1}$$

Then, we fitted a function to the speed-up for the best guess (green circles), which we derived from Amdahl's law.

$$\tilde{S}(N) = \frac{1}{s + \frac{p}{N}}, \quad s, p \in (0, 1) \tag{2}$$

$$\tilde{S}_{1536}(N) = \frac{\tilde{S}(N)}{\tilde{S}(1536)} = \frac{\tilde{T}_{1536}}{\tilde{T}_{\text{LOOP}}(N)} = \frac{t_{\text{s}} + \frac{t_{\text{p}}}{1536}}{t_{\text{s}} + \frac{t_{\text{p}}}{N}} = \frac{s + \frac{p}{1536}}{s + \frac{p}{N}} = \frac{s + \frac{1-s}{1536}}{s + \frac{1-s}{N}} \tag{3}$$

s denotes the serial portion and $p = 1 - s$ the parallel portion of the code. So, the only fit parameter is s and we are able to derive the serial portion of the problem. The corresponding results may be found in the right panel of Fig. 1.

For the present case, the serial portion is $s = 0.000023(8)$, so if 1536 cores ($\equiv$ 64 nodes) are employed for the job, the scale efficiency ($\eta = \tilde{S}(N)/N$) is still 97 %. Increasing the number of employed cores to 18 600 ($\equiv$ 775 nodes) results in a scale efficiency of $\sim$70 %. In summary, the code performed excellently on the Hazel Hen cluster on several 100 nodes.

We typically employ complete nodes, so that the number of cores is generally a multiple of 24. Concerning calculations on Hazel Hen, experience has shown that very long queue times are necessary for runtimes larger than 4–8 h. We have therefore optimized most of our jobs for a 4 h walltime, although longer runs would be desirable.

Memory Demand
The codes we typically use make use of shared memory, so the required amount of RAM does not grow strongly with the number of MPI processes. There is however one exception: When setting up k-point groups the problem has to be copied to each group, so the RAM amount increases almost linearly with the number of k-point groups (KPAR). Typical calculations of the size of test above needed about 35 GB RAM per computer node. This translates to 1430 MB per MPI task. On order to reduce the RAM demand, we can reduce KPAR, which increases the execution time, though.

3 Results

The results of our calculations are presented as follows: at first, we describe the investigated system and its band structure. This allows to relate morphological features with electronic states. In a second step, we investigate the H adsorption and the H-induced modification of the electronic states.

3.1 Pristine Si(557)-Au Surface

The Si(557) surface is a so called vicinal Si(111) surface, with an inclination of 9.5° from the (111) surface toward the $[\bar{1}\bar{1}2]$ direction [9, 26, 27]. Evaporation of 0.18 ML Au onto this surface leads to the formation of a single atomic Au strand per terrace. The (111) nanoterraces are 19.1 Å wide. We employ here the commonly accepted stable surface model proposed in Ref. [28] and shown in Fig. 2. According to this model, the Si(557)-Au surface consists of a single row of Au atoms that substitute Si surface atoms, and forms a honeycomb-like chain of Si reconstruction at the step edge. A Si adatom chain (in blue), surrounded by threefold coordinated restatoms

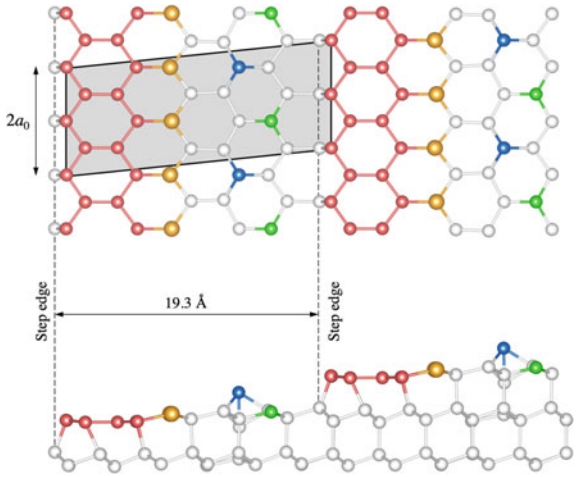


Fig. 2 Top and side view of the Si(557)-Au surface. Si honeycomb chain atoms are represented in red, Si adatoms in blue, Si restatoms in green, while other fourfold coordinated Si atoms are white. The Au chain is represented in yellow. The (5 × 2) unit cell is highlighted in gray.

(in green) runs parallel to the Au chain. Both the adatom and the restatom chains have twice the substrate periodicity and feature unoccupied orbitals. These make this surface chemically more reactive with respect to other similar surfaces [12, 28]. As a result, metallic surface states of different origin characterize the electronic band structure, as we will discuss in the following.

The model shown in Fig. 2 is used as a basis for the investigation of the H adsorption. Thereby, we use slabs consisting of 5 Si atomic bilayers and the surface termination within a (5 × 2), (5 × 4) or (5 × 8) periodicity (147, 294 and 588 atoms) to model different H coverages. Such large cells are needed both to model realistic H depositions and to identify concentration-dependent trends on the electronic band structure. We remark that systems of this size, in connection with the number of investigated configurations, can be reasonably modeled only in supercomputer systems such as Hazel Hen.

The bottom surface is saturated with hydrogen atoms. A vacuum region of about 11 Å separates periodic images of the slabs. A Γ-centered $4 \times y \times 1$ k-point mesh was used to carry out the integration in the Brillouin zone, with $y = 12$ for slabs with the (5 × 2) and (5 × 4) periodicity and $y = 3$ for slabs with the (5 × 8) periodicity. The first 3 Si bilayers (including H saturation) are frozen at the bulk positions, while the remaining atoms were allowed to relax until the forces acting on each atom were lower than 5 meV/Å.

The electronic band structure of the clean Si(557)-Au surface is shown in Fig. 3. We show only the path in the Brillouin zone corresponding to the Au chain direction in real space, i.e. the line joining the zone center (Γ) with the zone edge at Y'. The direction perpendicular to the wires indeed features nearly dispersionless bands [3]. The states with strong Si adatom, step edge or restatom character are colored correspondingly, according to the color coding defined in Fig. 2. According to previous calculations, the band structure of the Si(557)-Au surface is characterized by two

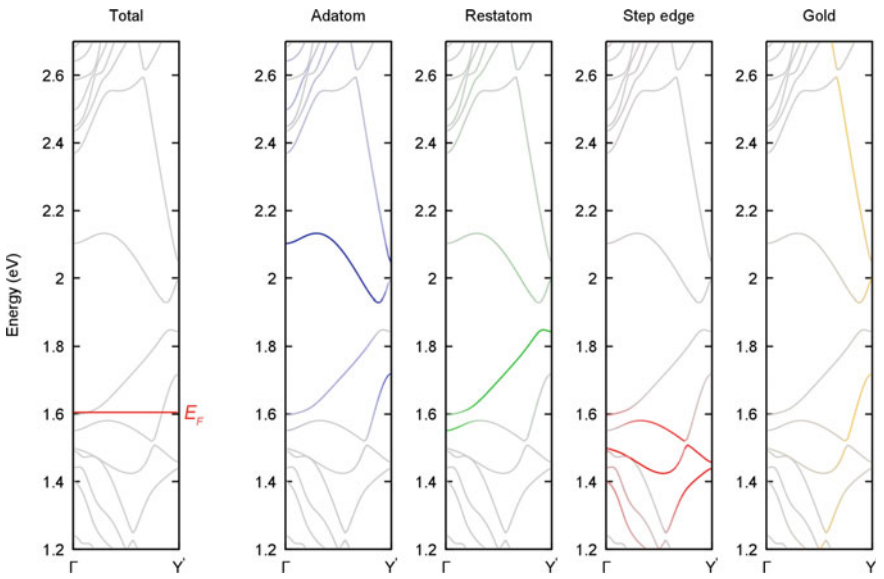


Fig. 3 DFT-PBE band structures of the clean Si(557)-Au surface modeled with a (5 × 2) surface unit cell. The left panel shows the total band structure. In the remaining panels the projection of the electronic states into atomic orbitals are represented. The surface projected bulk bands are shaded in gray and shown in the first panel. The color coding of the surface states denote their localization in real space: bands with strong Si adatom character are blue, bands with strong Si restatom character are green, while states related to Si step edge atoms are red.

states that cross the Fermi level [3, 29]. An Au-adatom hybrid band crosses the Fermi level close to the Brillouin zone boundary (blue and yellow band). A restatom band crosses the Fermi level with an extremely flat slope close to the Γ point (green band). The H adsorption can thus tune the system metallicity by saturation of one or more partially occupied orbitals.

In the outmost right panel of Fig. 3 the electronic states with strong Au character are highlighted. At the Y′ point, in the unoccupied part of the band structure, at about 1.9 eV a band gap between the Au states can be observed. This gap and its bridging, observed experimentally [10], plays an important role in our further argumentation (see below).

3.2 H Adsorbed Si(557)-Au

In order to investigate the impact of the H adsorption on the electronic structure of the Si(557)-Au surface, the adsorption configuration has to be determined at first. Therefore, we have calculated the potential energy surface (PES) for atomic H adsorption, which gives an approximate idea of the stable adsorption sites and a map

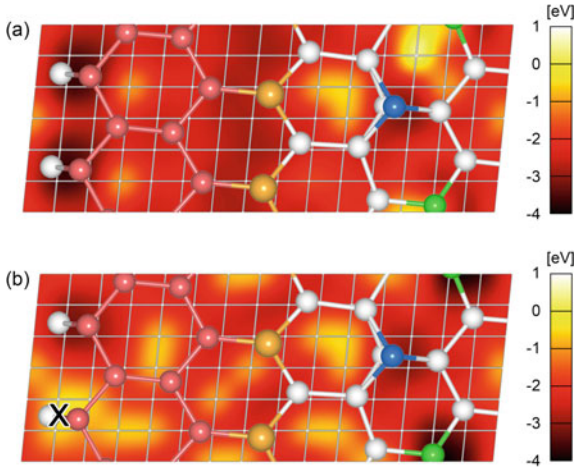


Fig. 4 (a) PES for the adsorption of atomic H on the clean Si(557)-Au surface. Darker regions mark favorable adsorption sites. The atomic structure of the surface termination is overlaid, color code as in Fig. 2. (b) PES for the adsorption of atomic H, assuming a H atom is already adsorbed at the Si step edge, in the position marked by a cross.

of the different energy minima on the surface. The PES is calculated constraining the surface and the lateral coordinates of the adsorbate, and allowing the remaining degrees of freedom to relax. We evaluate the corresponding energy for 90 possible positions in a regular raster, and interpolate the results for the positions in between.

The potential energy surface (PES) for the adsorption of atomic H on the clean Si(557)-Au surface is plotted in Fig. 4 (a). It shows a corrugated energy landscape, with global minima close to the Si step edge atoms and local minima close to the adatom and restatom chains. The PES furthermore suggests, that H adsorption on the Au chain is not energetically favorable.

Due to the constraints on the atomic relaxation, the PES does not represent the adsorption energy. The latter is obtained starting a structural optimization without any constraint from the minima of the PES. Thereby, we start with the adsorbate in several of the favored adsorption sites as obtained from the PES calculation. The adsorption energy at the lattice site X is then estimated as

$$E_{\text{ads}} = E_{\text{slab}}^{\text{H@Si(557)},X} - E_{\text{slab}}^{\text{Si(557)}} - E_{\text{gas}}(\text{H}). \tag{4}$$

Both the adsorption at the Si step edge ($E_{\text{ads}} = -4.55\,\text{eV}$) and at the restatom site ($E_{\text{ads}} = -4.65\,\text{eV}$) are very stable and characterized by a very similar energy. H deposition at the Si adatom also leads to a strong bond with the substrate ($E_{\text{ads}} = -3.91\,\text{eV}$). H adsorption close to the Au chain is about 1.5 eV less stable than adsorption at the restatom and therefore much less probable. Moreover, the H adsorption deeply influences the PES for the adsorption of a second H atom. The PES calculated for the adsorption of atomic H on the Si(557)-Au surface with H adsorbed, e.g., at the Si step edge is shown in Fig. 4 (b). The global minimum for the adsorption of a second H atom is clearly close to the restatom, with much shallower local minima at the Si step edge and at the Si adatom. Again, the H adsorption on the Au chain is not energetically favorable. The deep modification of the energy

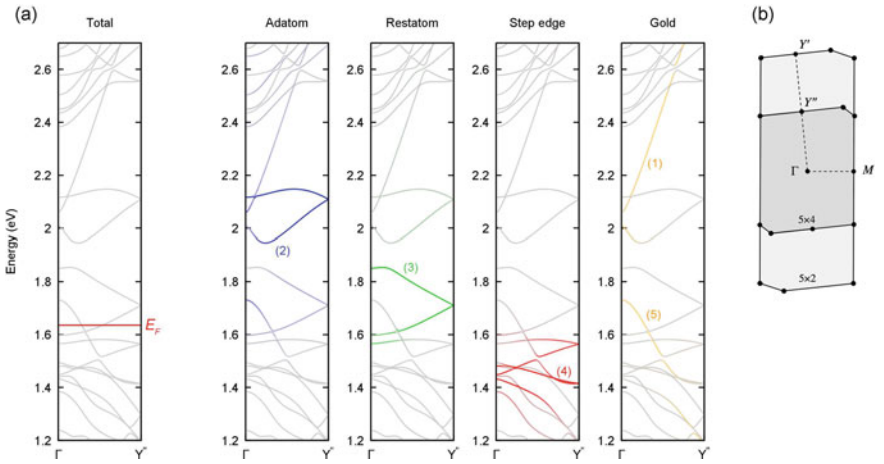


Fig. 5 (a) DFT-PBE band structures of the clean Si(557)-Au surface modeled with a (5 × 4) surface unit cell. The left panel shows the total band structure, while in the remaining panels the projection of the electronic states into atomic orbitals are represented. Color coding as in Fig. 3. (b) Surface Brillouin zone of the Si(557)-Au surface with (5 × 2) and (5 × 4) periodicity.

landscape upon H adsorption suggests that no ordered adsorption (e.g. at the Si step edge, until full coverage of the step edge) might be expected. From these calculations we conclude that atomic H is adsorbed close to the restatom site and at the Si step edge. However, since adsorption is a random process, and the H adsorption modifies the energy landscape, transient, i.e., metastable occupation of the adatom sites may be occupied with some probability. Adsorption at the gold chain, on the other hand, can be excluded, since the energy difference to the most favorable sites is large.

We now turn to the effect of the H adsorption on the band structure, and first discuss the influence of adsorption on selected surface sites. We use a unit cell of (5 × 4) periodicity to model the influence of the H adsorption on the electronic states, as it allows to model more realistic H concentrations. However, doubling the unit cell causes a folding of the band structure that makes its interpretation less direct. Figure 5 shows the band structure of the clean Si(557)-Au surface, calculated for the slab with doubled periodicity in the chain direction. Comparing the band structure with Fig. 3, both the band folding as well as the origin of the states can be recognized. Again, we observe (now at Γ, due to the folding) the relatively large band gap of 0.33 eV between the Au bands (labeled as (1) and (5) in Fig. 5). This gap would not allow the plasmon propagation measured in Ref. [16] without presence of the adatom and restatom states labeled by (2) and (3) in Fig. 5). Although these states do not close the gap, they reduce it to an extent that the life-time broadened bands overlap sufficiently to allow a continuous plasmon propagation [12]. In fact, they are also spatially close to the gold states and partially hybridize with them, so that they can act as a "bridge" for the plasmons, and reduce the effective gap to 0.09 eV.

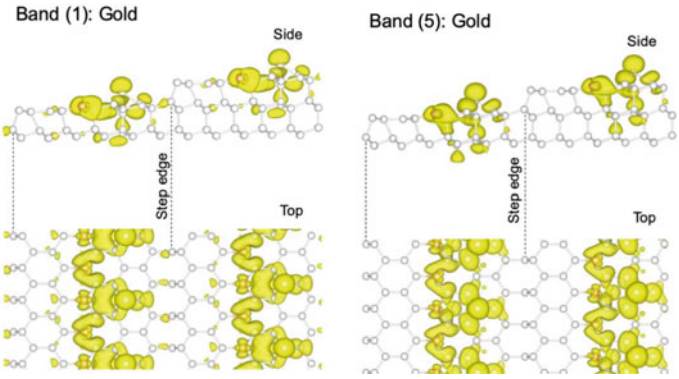


Fig. 6 Squared wavefunctions at the Brillouin-zone center associated with the Au bands (1) and (5) in Fig. 5.

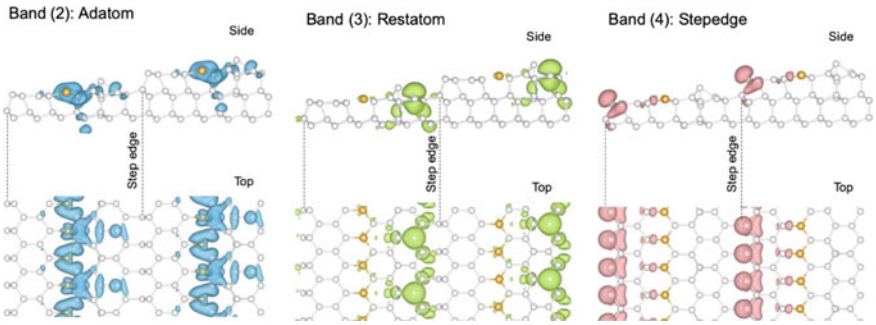


Fig. 7 Squared wavefunctions at the Brillouin-zone center associated with the bands (2), (3) and (4) in Fig. 5.

In order to underline these arguments, we plot in Figs. 6 and 7 the squared wavefunctions at the Brillouin-zone center associated with the bands (1) to (5). The two Au bands are strongly surface localized and very similar (Fig. 6), mirroring the fact that they are two parts of the same original band (in the (5×1)-periodicity), in which a gap is opened at the zone edge. It can be also observed that the bands we address as gold bands are indeed a hybrid involving Au and adatom states. The spatial extension of these states thus includes both the Au and the adatom-restatom chains. The only difference between band (1) and band (5) is a slightly different degree of hybridization with the adatom states.

In order to bridge the band gap for the plasmons, the electronic states not only need to be energetically within the band gap between the Au bands, they also have to spatially share the same region. Adatom and, to a lesser extent, restatom levels satisfy this requisite. Figure 7 shows the squared wavefunctions at the zone center associated with the bands (2) to (4). As mentioned, the adatom states strongly hybridize with the Au states, while hybridization with the restatom states is smaller. However, the

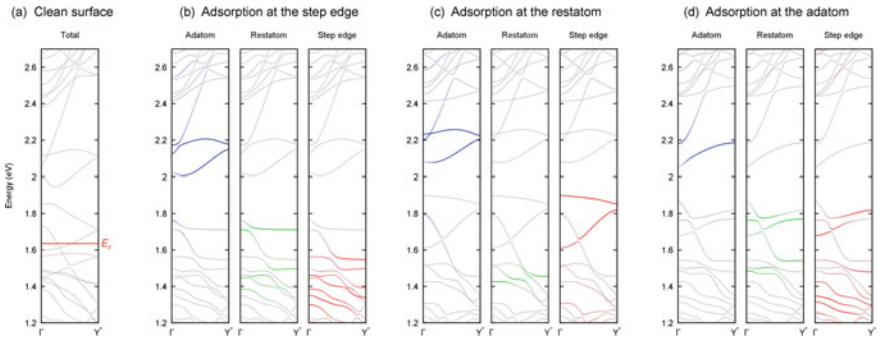


Fig. 8 (a) Electronic band structure calculated for clean Si(557)-Au surface using a (5 × 4) unit cell. (b) Band structure after adsorption of one H per (5 × 4) unit cell at the step edge, together with the projections of the bands onto the individual structural elements, indicated at the top. (c), (d) as in (b), after adsorption of the same amount of H at the restatom and the adatom chains.

step edge states are localized at a different surface region. Thus, they are not suited to bridge the gap between the Au bands, even if they would have a suitable energy.

The H adsorption at all investigated sites modifies the electronic band structure by electron transfer from H to the surface. This modifies mainly the occupation of states spatially close to the adsorption site. In turn, it has the principal effect to open (or widen) the band gap between the Au bands in the unoccupied part of the band structure and, most important, between adatom and restatom states. However, the magnitude of this effect is site specific, since different bands are involved, as shown in detail in Fig. 8.

Figure 8(a) reproduces the band structure of the clean surface, which is only shown for comparison. Panels (b), (c) and (d) show the modification of the bands of the clean surface upon hydrogenation of step edge, restatom and adatom chain, respectively. In order to emphasize hybridization and the changes introduced by atomic hydrogen on the single bands, we also show the main character of the bands close to E_F in the same color code as in Figs. 3 and 8. As seen from the energetic considerations described above, the occupation of restatom and step edge sites by atomic H should play the dominant role.

The by far largest effect on the band structure is due to H adsorption at the Si step edge atoms, shown in panel (b). Due to charge redistribution, not only the states related to the Si step edge, but also the states related to the restatom are shifted downwards, leading to the opening of the original, indirect gap of 0.092 eV between adatom and restatom states by about 0.171 eV. The occupation of this site alone, however, cannot explain the experimental finding that plasma frequency is strongly limited at large $k_{\|}$ [16].

H adsorption close to the restatom, illustrated in panel (c) of Fig. 8, has the main effect to shift downwards the bands related to the rest atoms. The electron transfer to the restatom bands is counterbalanced by an upward shift of the step edge bands, as the system has to remain charge neutral. Moreover, the adsorption at the restatom

site does not heavily affect the position and dispersion of the adatom bands and, due to the upwards shift of the step edge bands, the widening of the band gap is merely about 0.086 eV. However, there is a significant redistribution of electronic states: upon hydrogenation the character of the unoccupied band at about 1.9 eV (0.3 eV above E_F at Γ) switches from being mainly located at the restatom to being located at the step edge.

The consequence for the excitation of the plasmons is a frequency cut-off, because there is no overlap of this band with the states of the gold chain. Since the restatom site is the most favorable adsorption site, its occupation by H, together with further upward shifts of other bands, blocks the bridging function of the gap in the unoccupied states strongly hybridized with the gold chain. This explains why a clear upper boundary of plasmon excitation appears on the H covered surface [16].

Finally, H adsorption at the adatom causes an important electron transfer to an empty adatom band, which is downshifted in the valence band. Correspondingly, as the system has to remain neutrally charged, the filling of the previously occupied bands is reduced, resulting in an upward shift of the bands as shown in panel (d). Also in this case the widening of the band gap is minor, and amounts to 0.096 eV. Since this state is occupied only to a small fraction, it should not have any dominant influence on the excitation spectrum.

Common to all investigated cases is that the strongly dispersive Au bands are not really affected by the H adsorption. This is not surprising, as the adsorption occurs spatially separated from the Au chains, and as the metallicity of this system is not due to the Au chain.

In order to estimate the coverage dependence of the gap opening, we repeated the calculations with very large supercells of (5×8) periodicity, which allow to model the H adsorption stepwise by increments of 0.25 h per (5×2) unit cell. Increasing the H presence at equivalent surface sites shows that the opening of the band gap is directly proportional to the H coverage. In reality, the adsorption does not proceed in an ordered manner. Figure 4 (b) shows, e.g., that the restatom site is the favored site once an H atom is adsorbed at the Si step edge. In other words, an averaged and concentration-dependent effect of all the adsorption induced modifications of the band structure discussed so far should be expected.

The limiting case is represented by the adsorption of two H atoms per (5×2) unit cell, saturating each band crossing the Fermi energy for the clean surface. Modelling this case, e.g., with one H atom adsorbed at the step edge and one at the restatom, leads to the downward shift of all the half filled states related to the Si step edge and the rest atom. No band crosses the Fermi energy, and the system becomes insulating. This coverage, however, is never reached in experiments [16]. We estimate that in experiments a maximum coverage 1 h atom per (5×2) unit cell is achieved, which is close to the scenarios described above.

4 Conclusions

The impact of H adsorption on the electronic structure of the Si(557)-Au system has been investigated by atomistic calculations within DFT. Our study reveals that the influence of atomic hydrogen goes far beyond simple charge transfer.

As Si surface bands are strongly hybridized with those of Au, H adsorption on the energetically most favorable sites at the Si step edge and the restatom chain does not only cause significant shifts of bands with a tendency of widening band gaps in the unoccupied part of the band structure, but it also strongly changes the character of hybridization. The rehybridization causes redistribution not only of charge in real space and bands in reciprocal space, it also affects the spatial distribution of wave functions for the unoccupied states. Thus, the relevant bands lose their function as effective interpolation points for plasmons to bridge the gap in the Au-related band. These findings again underline the close coupling between unoccupied band structure and plasmonic excitations, and show how complex is the intertwinement between atomic nanowires and their environment.

Acknowledgements We gratefully acknowledge financial support from the Deutsche Forschungsgemeinschaft in the research unit FOR 1700 (project SA 1948/1-2). The Höchstleistungrechenzentrum Stuttgart (HLRS) is gratefully acknowledged for grants of high-performance computer time.

References

1. T. Giamarchi, *Quantum Physics in One Dimension* (Clarendon Press, Oxford, 2007)
2. P.C. Snijders, H.H. Weitering, Colloquium: electronic instabilities in self-assembled atom wires. Rev. Mod. Phys. **82**(1), 307 (2010)
3. F. Edler, I. Miccoli, J.P. Stöckmann, H. Pfnür, C. Braun, S. Neufeld, S. Sanna, W.G. Schmidt, C. Tegenkamp, Tuning the conductivity along atomic chains by selective chemisorption. Phys. Rev. B **95**(12), 125409 (2017)
4. S.D. Sarma, W. Lai, Screening and elementary excitations in narrow-channel semiconductor microstructures. Phys. Rev. B **32**(2), 1401–1404 (1985)
5. Z. Mamiyev, S. Sanna, C. Lichtenstein, T. Tegenkamp, H. Pfnür, Extrinsic doping on the atomic scale: tuning metallicity in atomic Au chains. Phys. Rev. B **98**(24), 245414 (2018)
6. Z. Mamiyev, M. Tzschoppe, C. Huck, A. Pucci, H. Pfnür, Plasmon standing waves by oxidation of Si (553)-Au. J. Phys. Chem. C (2019)
7. T. Kanagawa, R. Hobara, I. Matsuda, T. Tanikawa, A. Natori, S. Hasegawa, Anisotropy in conductance of a quasi-one-dimensional metallic surface state measured by a square microfour-point probe method. Phys. Rev. Lett. **91**(3), 036805 (2003)
8. T. Nagao, G. Han, C. Hoang, J.-S. Wi, A. Pucci, D. Weber, F. Neubrech, V.M. Silkin, D. Enders, O. Saito, M. Rana, Plasmons in nanoscale and atomic-scale systems. Sci. Technol. Adv. Mater. **11**(5), 054506 (2010)
9. A. Kirakosian, R. Bennewitz, J.N. Crain, Th. Fauster, J.L. Lin, D.Y. Petrovykh, F.J. Himpsel. Atomically accurate Si grating with 5.73 nm period. Appl. Phys. Lett. **79**(11), 1608–1610 (2001)
10. T. Lichtenstein, Z. Mamiyev, S. Sanna, W.G. Schmidt, C. Tegenkamp, H. Pfnür, Probing quasi-1D band structures by plasmon spectroscopy. Phys. Rev. B **97**, 165421 (2018)

11. T. Lichtenstein, Z. Mamiyev, E. Jeckelmann, C. Tegenkamp, H. Pfnür, Anisotropic 2D metallicity: plasmons in Ge(100)-Au. J. Phys. Condens. Matter **17**(31), 175001 (2018)
12. S. Sanna, T. Lichtenstein, Z. Mamiyev, C. Tegenkamp, H. Pfnür, How one-dimensional are atomic au chains on a substrate? J. Phys. Chem. C **122**(44), 25580–25588 (2018)
13. J.N. Crain, Low-dimensional electronic states at silicon surfaces. Appl. Phys. A **82**, 431–438 (2006)
14. T. Nagao, S. Yaginuma, T. Inaoka, T. Sakurai, One-dimensional plasmon in an atomic-scale metal wire. Phys. Rev. Lett. **97**(11), 116802 (2006)
15. S.C. Erwin, P.C. Snijders, Silicon spin chains at finite temperature: dynamics of Si(553)-Au. Phys. Rev. B **87**(23), 235316 (2013)
16. Z. Mamiyev, S. Sanna, F. Ziese, C. Dues, C. Tegenkamp, H. Pfnür, Plasmon localization by h-induced band switching. J. Phys. Chem. C **124**(1), 958–967 (2020)
17. J.P. Perdew, W. Yue, Accurate and simple density functional for the electronic exchange energy: generalized gradient approximation. Phys. Rev. B **33**, 8800–8802 (1986)
18. G. Kresse, J. Furthmüller, Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set. Comput. Mater. Sci. **6**(1), 15–50 (1996)
19. G. Kresse, J. Furthmüller, Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. Phys. Rev. B **54**, 11169–11186 (1996)
20. P.E. Blöchl, Projector augmented-wave method. Phys. Rev. B **50**, 17953–17979 (1994)
21. G. Kresse, D. Joubert, From ultrasoft pseudopotentials to the projector augmented-wave method. Phys. Rev. B **59**, 1758–1775 (1999)
22. J.P. Perdew, K. Burke, M. Ernzerhof, Generalized gradient approximation made simple. Phys. Rev. Lett. **77**, 3865–3868 (1996)
23. G.I. Csonka, J.P. Perdew, A. Ruzsinszky, P.H.T. Philipsen, S. Lebègue, J. Paier, O.A. Vydrov, J.G. Ángyán, Assessing the performance of recent density functionals for bulk solids. Phys. Rev. B **79**, 155107 (2009)
24. J.P. Perdew, A. Ruzsinszky, G.I. Csonka, O.A. Vydrov, G.E. Scuseria, L.A. Constantin, X. Zhou, K. Burke, Restoring the density-gradient expansion for exchange in solids and surfaces. Phys. Rev. Lett. **100**, 136406 (2008)
25. C. Braun, C. Hogan, S. Chandola, N. Esser, S. Sanna, W.G. Schmidt, Si(775)-Au atomic chains: geometry, optical properties, and spin order. Phys. Rev. Mater. **1**, 055002 (2017)
26. M. Henzler, The step structure of the Si(557) surface. Thin Solid Films **428**(1–2), 129–132 (2003)
27. I.K. Robinson, P.A. Bennett, F.J. Himpsel, Structure of quantum wires in Au/Si(557). Phys. Rev. Lett. **88**(9), 096104 (2002)
28. J.N. Crain, J. McChesney, F. Zheng, M.C. Gallagher, P. Snijders, M. Bissen, C. Gundelach, S.C. Erwin, F.J. Himpsel, Chains of gold atoms with tailored electronic states. Phys. Rev. B **69**(12), 125401 (2004)
29. H.S. Kim, S.Y. Shin, S.H. Uhm, C.C. Hwang, D.Y. Noh, J.W. Chung, Evidence of metallic nature of the surface bands of Au/Si(557). Phys. Rev. B **80**, 033412 (2009)

Anderson Transitions and Interactions

Final Status Report

Matthias Stosiek, Felix Weiner, and Ferdinand Evers

Abstract In this report we present our study of disordered interacting systems. Our interest here is two-fold: we want to study superconductor-insulator-transition in disordered thin film superconductors and the many-body localization transition in interacting one-dimensional wires. Our main topic, disordered thin film superconductors, were modelled within a mean-field framework. We investigate the effect of full self-consistency (sc) on local observables within the Boguliubov-deGennes (BdG) theory of the attractive-U Hubbard model in the presence of on-site disorder; the sc-fields are the particle density $n(\mathbf{r})$ and the gap function $\Delta(\mathbf{r})$. For this case, we reach system sizes unprecedented in earlier work. They allow us to study phenomena emerging at scales substantially larger than the lattice constant, such as the interplay of multifractality and interactions, or the formation of superconducting islands. For example, we observe that the coherence length exhibits a non-monotonic behavior with increasing disorder strength already at moderate U. In our study of the many-body localization (MBL) transition, we integrate the Schrödinger equation exactly by means of a Chebyhsev expansion of the time evolution operator. The dynamics of such systems is probed via a density-density correlation function. Careful analysis of our numerical data gives evidence that the MBL-phase splits into two subphases. In addition to the conceptual relation of the two topics we profit from shared methodology in our kernel polynomial method(KPM) approach in describing these systems.

1 Scientific Work Accomplished and Results Obtained

The project that has been performed here was a very ambitious one. The reason is twofold. On the computational side we implemented substantial optimizations of the Chebyshev expansion. On the physics-side that data analysis was highly complicated, because the strong system size dependence had to be understood before statements

M. Stosiek · F. Weiner · F. Evers (✉)
Fakultät für Physik, Institur für Theoretische Physik, Universität Regenburg, Universitätsstraße 31, 93053 Regensburg, Germany
e-mail: ferdinand.evers@physik.uni-regensburg.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_10

on the thermodynamic limit became feasible. Only our massively parallel setup has allowed us to achieve the very large sample numbers required to do this extrapolation reliably. As a consequence, only relatively recently we were able to harvest the results that go into several publications [1–4].

In order to give evidence for the status of the computational methodology developed in this project, we would like to mention that in recent time we have been asked to join two new collaborations, with Prof. Burmistrov (Landau Institute) and Prof. Wulfhekel (KIT), that embark on our achievements.

In the interest of brevity, we report a selection of the results that were already published and the ones that will be published in the near future.

A detailed introduction to our model and methodology can be found in our recently published papers on disordered thin film superconductors [1] and disordered interacting 1D wires [2].

1.1 Superconductors with a Single Impurity

In a collaboration with the Wulfhekel group, we study the response of the superconducting gap to a Fe impurity on the surface of an Al(111) surface. The Wulfhekel group was conducting STM experiments, whereas we provided theoretical insight employing numerical simulations. Numerical simulations are necessary, as analytical formalisms are only available for specific cases [5], due to the complications brought about by the self-consistency requirement in the mean-field description of a superconducting system. We find an excellent agreement in the response at the impurity site and provide an explanation in terms of the density response in the normal state [4].

1.2 Enhancement of the Critcal Temperature by Disorder

The enhancement of superconductivity by disorder has long been predicted analytically within a partial self-consistency approach [6, 7]. Recently this effect could be demonstrated experimentally as well [8]. We for the first time find a substantial increase in the zero temperature gap in a fully self-consistent approach in the mean-field description of the Hubbard model. In Fig. 1a clear local maximum for intermedia disorder strength ($W = 1.25$) can be seen. The mean-field critical temperature is dramatically enhanced with respect to the clean critical temperature T_c^{BCS}.

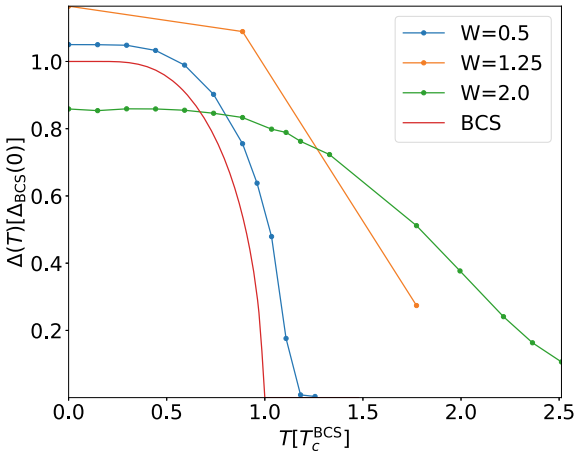


Fig. 1 Temperature (T) dependance of the disorder-averaged mean superconducting gap for various values of disorder (W). (Parameters: $n = 0.3$, $U = 2.2$).

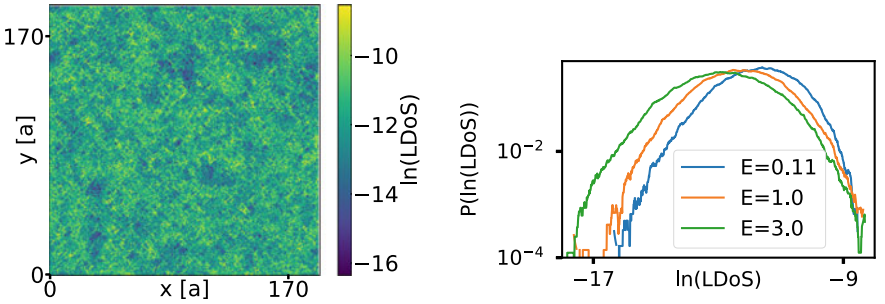


Fig. 2 Distribution of the local density of states (LDoS), $\rho(E, \mathbf{r})$. Left: Spatial distribution for a typical sample at the coherence peak energy of DoS ($E = 0.11$, $L = 96$) Right: Corresponding distribution function of LDoS at energies $E = 0.11$, 1.0, 3.0 illustrating the flow of the distribution with E. (Parameters: $W = 1.5$, $U = 1.5$, $L = 96$, $n = 0.875$).

1.3 Mesoscopic Fluctuations of LDoS and Local Gap Function

LDoS Distribution

To characterize the statistical properties of the local density of states and the local gap function we analyzed distribution and autocorrelation functions [1]. We compared numerical findings with predictions from analytical theories. In Fig. 2 (left) we display a spatial map of the LDoS, $\rho(E, \mathbf{r})$, of a typical sample with moderate disorder at interaction $W \gtrsim U \gtrsim 1$ and linear size $L = 96$ lattice constants. The logarithmically broad distribution is readily identified. The log-normal shape of the distribution is already known from non-interacting disordered systems [9]. The corresponding distribution function is displayed in Fig. 2 (right).

The LDoS-distribution has been investigated analytically by the team based at the Landau-Institute [10]. At temperatures above the critical temperature T_c. Our observations are broadly consistent with these results, since it is reported that the distribution develops a pronounced non-Gaussian character upon decreasing the temperature.

LDoS Fluctuations

In a collaboration with Igor Burmistrov of the Landau Institute we compare an analytical theory for the variance of the LDoS with our fully self-consistent numerical simulations. We find an excellent agreement between our simulations and the analytical description in the low disorder limit, where it is applicable to [3].

Autocorrelations of Gap Function and Coherence Length

We investigated the disorder averaged spatial autocorrelator $\Phi_{\mathrm{lg}}(\mathbf{q}) = \overline{|\Delta(\mathbf{q})|^2}$ of the pairing function $\Delta(\mathbf{r})$ in Fourier space. Notwithstanding anisotropy at $q \approx a^{-1}$, in the limit of small wavenumbers $q \ll a^{-1}$ the correlator $\Phi_{\mathrm{lg}}(q)$ is isotropic and with good accuracy we have

$$\frac{\Phi_{\mathrm{lg}}(0)}{\Phi_{\mathrm{lg}}(q)} = 1 + (q\xi)^2 + \ldots \tag{1}$$

where $\Phi_{\mathrm{lg}}(0) := \Phi_{\mathrm{lg}}(q \to 0)$. Both the increase of $\Phi_{\mathrm{lg}}(0)$ (as approximated by $\Phi_{\mathrm{lg}}(\pi/L, 0)$) with disorder and the characteristic length ξ have been displayed in Fig. 3 (right).

$\xi_U(W)$ exhibits a local non-monotonicity on its way from the clean to the dirty limit; the non-monotonic decay is readily seen also from the original data Fig. 3 (left). This peculiar behavior should be interpreted in connection with the formation of superconducting islands. It occurs in the same parameter range and may relate to a percolation transition. We mention that the non-monotonous shape, Fig. 3 right, has been observed before in [11], however, at unrealistically strong interactions $U = 5$. Our work shows that island formation carries over all the way into the physically relevant regime of intermediate parameter values.

We therefore have demonstrated in this work that an understanding of the superconductor-insulator transition (SIT) as it is observed in experiments must include the formation of islands as an indispensable ingredient. In present analytical studies, the effect is still ignored.

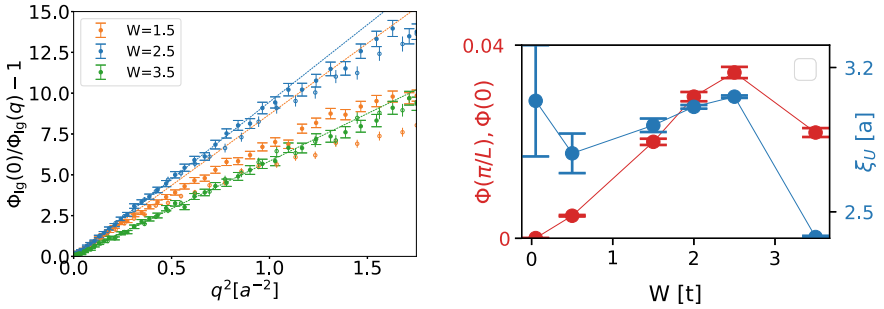


Fig. 3 Left: The non-trivial part of the inverted normalized gap autocorrelation function $\Phi_{lg}^{-1}(\mathbf{q}) = \overline{|\Delta(\mathbf{q})|^2}^{-1}$ evolving with W at fixed U. Φ_{lg} is shown along directions $(\pi/a, 0)$ (full symbols) and $(\pi/a, \pi/a)$ (open). (Parameters: $U = 1.5$, $L = 192$) Right: Variation of $\Phi_{lg}(\pi/L, 0)$ and $\Phi_{lg}(0)$ (red) and the correlation length, ξ (blue) with increasing disorder (left, $U = 1.5$). $\Phi_{lg}(0)$ coincides with $\Phi_{lg}(\pi/L)$ within the symbol size as portrayed here. (Parameters: $L = 192$)

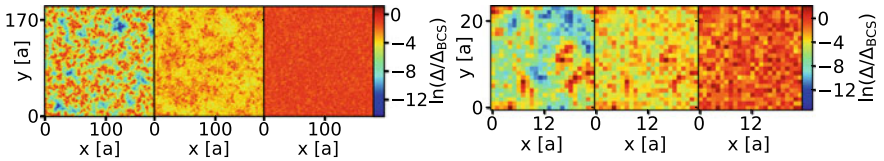


Fig. 4 The spatial distributions of the pairing amplitude as calculated from different self-consistency schemes are compared. Left: full self-consistency. Center: energy-only self-consistency with inhomogeneous Hartree shift. Calculation is done with the single-particle ("screened") potential as it results from the full scf-calculation, left. Right: energy-only scheme. Left: Typical 2D sample for high disorder. (Parameters: $U = 1.5$, $W = 3.5$) Right: 2D slice of a typical 3D sample at the Anderson transition. (Parameters: $U = 2.5$, $W = 4.0$)

1.4 Impact of Self-consistency

Effects of Self-Consistency Schemes on the Local-Gap Distribution

We compared the results of full and energy-only self-consistency schemes for the local pairing amplitude $\Delta(\mathbf{r})$ for the Anderson Problem in 2D and 3D. Energy-only self-consistency refers to a approximate self-consistency scheme, in which only the energies but not the eigenfunctions are renormalized during the self-consistency cycle [1]. Since localization is sensitive to spatial dimensionality, we discuss 2D and 3D separately.

2D

Figure 4 (left) shows a spatial distribution of $\Delta(\mathbf{r})$ as obtained for typical 2D sample at intermediate interaction and high disorder values. The calculation with full self-consistency, Fig. 4 (left) exhibits a clear tendency towards the formation of superconducting islands. In contrast, with energy-only self-consistency, a rather homogeneous speckle pattern is found missing any indications of island formation. Hence,

already by inspecting individual samples we expect that distribution functions of physical observables will depend in a qualitative way on the applied scf-scheme in broad parameter regions. In order to highlight the effect of screening, we have displayed in Fig. 4 also the results of an intermediate scf-scheme. It operates in an energy-only mode, but adopts for the disorder the effective single particle potential ("screened" potential) as it is obtained as a result from the full scf-calculation. As is seen from Fig. 4 (left,) first indications of islands emerge, but the contrast is still largely underestimated. This result underlines the importance of full consistency in the scf-procedure.

3D

Figure 4 (right) shows a 2D slice of the spatial distribution of $\Delta(\mathbf{r})$ as obtained for typical 3D sample at intermediate interaction and intermediate disorder. The parameters are chosen, so that the corresponding non-interacting Anderson problem is critical. As in the 2D case, the field obtained within the fully self-consistent calculation shows a pronounced formation of islands. The energy-only scheme in analogy to our 2D results exhibits a rather homogeneous spatial distribution. The results of the energy-only scheme with "screened" potential again show first indications of island development with dramatically underestimated contrast. This highlights the importance of full self-consistency also in 3D.

To what extent the conclusions of earlier theoretical works that consider this scenario [6, 12] are affected remains to be seen.

Effects of Self-consistency on Gap Autocorrelator

Figure 5 shows data analogue to Fig. 3, now with energy-only self-consistencies.

The most striking and perhaps unexpected feature is a qualitative difference. In the full scf-calculation, $\Phi_{\mathrm{lg}}(q)$ follows Eq. (1) and exhibits a well defined parabolic shape in the vicinity of small wavenumbers. This feature is not reproduced within

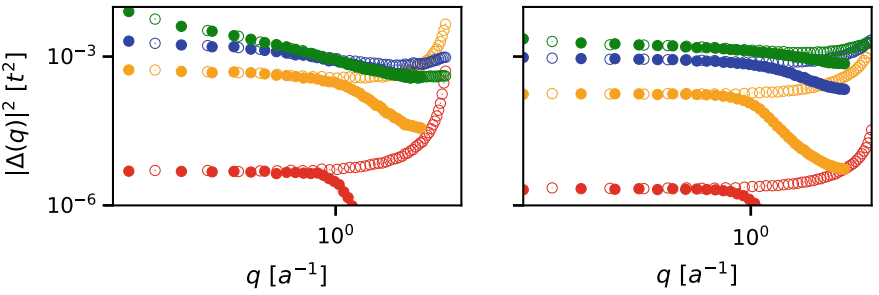


Fig. 5 Gap autorcorrelation function $\Phi_{\mathrm{lg}}(\mathbf{q}) = \overline{|\Delta(\mathbf{q})|^2}$ calculated employing two different energy-only self-consistency schemes. Φ_{lg} is shown along directions $(\pi/a, 0)$ (full symbols) and $(\pi/a, \pi/a)$ (open); traces for four different disorder values are shown, $W = 0.05, 0.5, 1.5, 2.5$, from bottom to top. Left: energy-only self-consistency with screened potential. Right: energy-only self-consistency. (Parameters: $U = 1.5$ $L = 96$)

energy-only schemes. The bare scheme does not exhibit an appreciable curvature up to $q \approx a^{-1}$, so $\xi \approx 0$. In contrast, within the screened scheme $\mathscr{P}_{\mathrm{lg}}$ does not show clear saturation at small wavenumbers within the range of q-values accessible. We thus interpret these results as a strong indications that wavefunction renormalization as it occurs within the full scf-scheme is crucial for understanding those aspects of qualitative physics that hinge on long-range spatial correlations.

1.5 Dynamical Signatures of the Many-Body Localization Transition

The second part of this project was concerned with the study of the so-called Many-Body Localization (MBL) transition. The MBL transition has attracted considerable attention over the last decade due to its novel character, which is very different from conventional equilibrium phase transitions or Anderson transitions on finite-dimensional lattices (see e.g. [13] for a review). The understanding of its critical properties is far from complete. From a computational point of view, the difficulty lies in solving the Schrödinger equation for many-body problems. As the corresponding Hilbert spaces grow exponentially with the number of particles, it is very demanding to obtain exact numerical results for large systems. Within the present project, sparse-matrix techniques have been used to obtain state-of-the art numerical data. Such data allowed us to present a comprehensive analysis of the finite-size effects with important conclusions regarding the dynamical signatures of the transition and the critical disorder strength.

The prototypical models for studying this phenomenon are one-dimensional quantum lattice models with a random contribution to the potential, which models the disorder in the system. It is well understood that, in the absence of interactions, such models are fully localized already for an arbitrarily small degree of randomness. This means that quantum particles cannot propagate through such a system, irrespective of their energy. The situation changes, however, if the particles interact among each other. Then, a large body of evidence has been accumulated in favour of the existence of a phase transition, the Many-Body Localization transition. This transition separates an ergodic/metallic phase at weak disorder and localized phase at strong disorder. The results obtained within this project aim at understanding the dynamical signatures of this transition. The reason for studying dynamical quantities is twofold:

1. the observables evaluated here are related to experimentally measurable quantities, such as the conductivity of the system
2. dynamical simulations can be performed for larger system sizes as compared to the evaluation of spectral quantities, such as eigenstate statistics. This is due to the existence of memory-efficient sparse-matrix algorithms for such problems.

The latter point is important because of the strong finite size effects that plague virtually all computational studies of the MBL transition. The strong finite-size effects

that are presence in dynamical quantities have been already examined in our previous work focusing on the ergodic, i.e. weakly disordered side of the transition [14]. With the help of computing resources on ForHLR2, we have extended these results focusing on larger disorder strengths. In this regime, the level spacing distribution of the system is close to Poissonian, which is widely seen as a signature of many-body localization. We demonstrate in Ref. [2] that a slow dynamics is present in the system at such disorder strengths. More specifically, we study the width $\Delta x(t)$ of a certain correlation function. $\Delta x(t)$ can be interpreted as a root mean-squared displacement corresponding to the spreading of excess charge on top of a thermal background. In a (many-body) localized system, it is expected that $\Delta x(t)$ reaches a value at long times $\Delta x(t \to \infty) = \xi$, where ξ is a measure of the localization length. However, we cannot observe such a saturation of $\Delta x(t)$ on the largest system sizes available. Instead, $\Delta x(t)$ diverges in time but with a growth weaker than any power. Based on this observation, we propose a scenario involving an intermediate phase, which we refer to as $\mathrm{MBL_A}$ in contrast to the actual many-body localized phase, $\mathrm{MBL_B}$. We would like to emphasise that the difference between the two phases can only be observed on sufficiently large systems, which requires considerable computational effort. In Ref. [2], we show results for system sizes up to $L = 32$ sites, corresponding to a Hilbert space dimension of $\sim 6 \cdot 10^8$.

2 Realization of the Project

Simulations Performed, Codes Used

To compute our self-consistent fields, we devised an algorithm based on the kernel polynomial method (KPM) [15]. With a KPM based code, we were able to reduce the asymptotic runtime from a cubic to a quadratic dependence on the size of the lattice compared to conventional full-diagonalization approaches. Through this we were able to achieve system sizes unprecedented in the literature. We achieved an approximately ideal inter-node parallelization both through the computation of multiple impurity configuration and also the computation of the self-consistent fields on different points in space with negligible communication between processes. A benchmark of the intra-node parallelization is shown in Fig. 6 (left). Our code is written in a combination of Python and C, combining the performance of a highly optimized C-kernel with the adaptability and convenience of a high-level language for the non performance critical sections. It is computationally expensive to find a single solution of the self-consistent fields ($\approx$100–1000 core-h) for one disorder configuration. We need a large number ($\approx$100000) of these configurations to achieve a reliable disorder average and to investigate different regimes of disorder strength, interaction strength, filling fraction and system size. In total we used 9,672,794 core-h. The full extent of the 10 mio core-h was not used so far, because of the service interruption by the security incident at the SCC. We produced approximately 1 TB of self-consistent

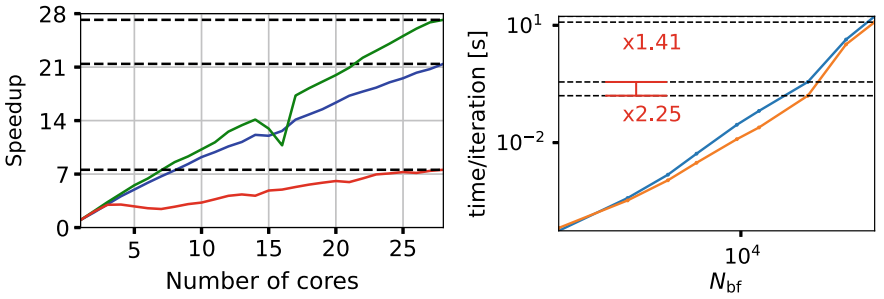


Fig. 6 Benchmarking intra-node parallelization and code performance. N_{bf} denotes the number of basis functions of the underlying system. Left: Speedup with the number of cores per process for different system sizes. The performance dips (green, blue: near 16; red: near 7,14,21,25) with rising number of cores we assign to a hardware issue related to caching. (Parameters: $N_{bf} = 18432$ (blue), $N_{bf} = 73728$ (green), $N_{bf} = 165888$ (red).) Right: Performance check comparing the matrix-free implementation (orange) with standard mkl_sparse_d_mm of the MKL Sparse BLAS library (blue). One iteration corresponds to one sparse matrix-vector product. The ratio of the timings of the MKL and matrix-free algorithms is shown in red at $N_{bf} = 73728, 294912$.

mean-field Hamiltonians across the parameter and impurity configuration space. We generated ≈20000 hdf5 files, where multiple parameter configurations can be saved in the same file for the same impurity configuration. For a typical job we used ≈5000 cores.

Code Modifications

To speed up a single self-consistency iteration we optimized the Chebyshev expansion during the project. Its performance critical part constitutes of the recursive action of the Hamiltonian on a basis vector. An implementation of the sparse-matrix vector product custom-tailored to our system is crucial for an optimal performance. The sparse-martrix vector multiplication is memory-bound, i.e. the performance is limited by the time it takes to fetch data from memory. For this reason we devised a self-written "matrix-free" matrix vector product that outperforms standard state of the art sparse-matrix vector multiplication libraries.

The idea is the following: Conventional sparse matrix packages keep all non-zero elements, i.e. value and index, in memory. Matrix-free implementations become efficient if many of the non-zero elements have identical values storing only the different values that occur.

With matrix-free implementations the graph of the Hamiltonian has to be hard-coded in the matrix-vector product routine. For our Hamiltonian the amount of memory load operations of matrix data is reduced by a factor of 6 reflecting the number of non-zero elements per row of our matrix. In addition, the integer indices corresponding to the matrix graph do not have to be loaded. Altogether, this leads to a

reduction of data to be loaded by a factor of 9. The datatype for values is double and for the indices is integer. Note, that the speed-up to be expected from the matrix-free implementation is less than a factor of 9. This is because not only the matrix but also the basis vectors have to be loaded from memory, so the reduction of memory load operations also depends on how many basis vectors are acted on in parallel. A benchmark comparing a conventional implementation with the MKL library and our matrix-free implementation can be found in Fig. 6 (right).

3 Third-Party Funded Projects that Used ForHLR2 to a Relevant Degree

DFG-funded projects: EV30/11-1; EV30/12-1; SFB-1277 (Projects A03 and B01). Future use by RFBR-DFG.

4 Co-operations that Used ForHLR2

4.1 Co-operations with External Partners

Prof. I. Burmistrov (Landau Institute of Theoretical Physics); Prof. W. Wulfhekel (Institute of Physics and Institute of Nanotechnology, KIT); Prof. B. Lang (Institute of Applied Informatics, BU Wuppertal).

4.2 PhD Theses Completed Within the Project

Matthias Stosiek (MSc) at University of Regensburg, submitted June 2020, defended July 2020.

References

1. M. Stosiek, B. Lang, F. Evers, Phys. Rev. B **101**, 144503 (2020)
2. F. Weiner, F. Evers, Phys. Rev. B. **100**, 104204 (2019)
3. M. Stosiek, F. Evers, I.S. Burmistrov, arXiv:2107.06728
4. M. Stosiek, C. Baretzky, T. Balashov, F. Evers, W. Wulfhekel, arXiv:2107.01646
5. A.L. Fetter, Phys. Rev. **140**, A1921 (1965)
6. M.V. Feigel'man, V.E. Kravtsov, L.B. Ioffe, E. Cuevas, Ann. Phys.-New York **325**, 1390 (2010)
7. J. Mayoh, A.M. García-García, Phys. Rev. B **92**, 174526 (2015)
8. K. Zhao et al., Nat. Phys. **15**, 904–910 (2019)
9. A.D. Mirlin, Phys. Rep. **326**, 259 (2000)

10. I.S. Burmistrov, I.V. Gornyi, A.D. Mirlin, Phys. Rev. B **93**, 205432 (2016)
11. G. Seibold, L. Benfatto, C. Castellani, J. Lorenzana, Phys. Rev. B **92**, 064512 (2015)
12. M.V. Feigel'man, L.B. Ioffe, V.E. Kravtsov, E.A. Yuzbashyan, Phys. Rev. Lett. **98**, 027001 (2007)
13. R. Nandkishore, D.A. Huse, Ann. Rev. Condens. Matter Phys. **6**, 15 (2015)
14. S. Bera, F. Giuseppe De Tomasi, F. Weiner, F. Evers, Phys. Rev. Lett. **118**, 196801 (2017)
15. A. Weiße, G. Wellein, A. Alvermann, H. Fehske, Rev. Mod. Phys. **78**, 275 (2006)

Materials Science

Johannes Hötzer and Britta Nestler

This chapter contains demonstrations of research results obtained from efficient simulation software solutions with high scalability properties for different application fields. Materials and processes are to be designed and optimized in an accelerated manner through the use of high performance simulations. The applications include battery materials, tribological systems and the prediction of skeletal muscles motions.

The first paper illustrates research results for the design of new electrode materials for lithium and post-lithium ion batteries based on highly parallelized quantum mechanical calculations using the density functional theory (DFT) code VASP. The first part of this contribution deals with Fe- and Co-doped CeO_2 as new insertion-type electrode in order to achieve fast charging properties, high volumetric and gravimetric capacity. The second part, density functional theory computations are performed to investigate intercalation mechanisms in aqueous Zn batteries. The computations determine a phase diagram in an electrochemical environment.

In the second contribution, large scale atomistic simulations are used to investigate nearsurface grain refinement in tribologically sheared metallic sliding surfaces. The simulations shed light on mechanisms of refinement and coarsening of the grain structure evolution and on the dynamics of these processes near the surface. A reduction in grain size near the surface yields hardening and material flow is minimized.

The last project gives insights into simulations of skeletal muscles as a neuromuscular system. The simulation methods involve multi-physics and multi-scale models which are solved on high performance computing architectures. The model

J. Hötzer · B. Nestler
Institute for Applied Materials - Computational Materials Science (IAM-CMS)
Karlsruhe Institute of Technology (KIT)
Straße am Forum 7, 76131 Karlsruhe, Germany
e-mail: britta.nestler@kit.edu

combines different physical, biochemical and electrical-signal processes on temporal and spatial scales. Optimized scalability and runtime is achieved by choosing appropriate compilers, by structuring the system of equations and by elaborating AVX vectorization techniques for code generation of CPUs.

Projects face the difficulty that some results are not completed and not fully proven due to the substantially reduced availability of Hazel Hen and Hawk during the project duration time. Verifications of the performance are hence the goal of forthcoming research.

Atomistic Modelling of Electrode Materials for Lithium and Post–lithium Ion Batteries

Holger Euchner

Abstract The first part of this report is about Fe- and Co-doped CeO_2 as new insertion–type electrode for lithium and post–lithium ion batteries. Alternative anode materials with high volumetric and gravimetric capacity, moreover, allowing for fast charging are of utmost importance for next-generation lithium and post-lithium ion batteries. Keeping this in mind, we have investigated a new insertion-type electrode material, metal-doped CeO_2. While CeO_2 shows limited capacity, the introduction of a small fraction of carefully selected dopants, results in a significant capacity increase. Interestingly, this capacity increase is accompanied by an off-centering of the dopant in the crystalline lattice and, moreover, results in the dopant atoms being reduced to their metallic state under alkaline metal insertion.

In the second part of this report the H^+/Zn^{2+}–intercalation in V_2O_5 for aqueous Zn batteries is discussed. Vanadium oxides are a promising class of cathode materials for aqueous zinc metal batteries with a still debated intercalation mechanism. To corroborate the experimentally observed intercalation of both H^+ and Zn^{2+} into δ-$Ca_{0.24}V_2O_5$ (CVO) and to furthermore identify a possible H^+/Zn^{2+}-exchange intercalation, this phase was investigated by density functional theory (DFT). For this purpose, the concept of the computational hydrogen electrode (CHE) was exploited to determine the phase diagram in an electrochemical environment.

1 Introduction to the First Part

The steadily increasing demand for better battery materials calls for the investigation of new electrode materials [1, 2]. To achieve rapid charging, this also aims at replacing graphitic anodes, with materials that have high capacity, high structural stability and fast kinetics [3–5]. These requirements may be best met by insertion-type materials such as CeO_2. This compound has not been investigated in detail, which is due to the increased price as compared to e.g. titanates and, moreover, due to the rather limited

H. Euchner (✉)
Helmholtz Institut Ulm for Electrochemical Energy Storage, Helmholtzstr. 11,
89081 Ulm, Germany
e-mail: holger.euchner@uni-ulm.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_11

capacity (about 155 mAh/g for a maximum lithium uptake of one Li per CeO_2). Surprisingly, the introduction of carefully selected metal dopants was shown to result in a dramatic increase in capacity by more than 200%. However, the reasons for this capacity increase were so far unclear and have been tackled by a computational approach.

2 Computational Methods

For quantitative evaluation of Fe-free and Fe-doped CeO_2, we have conducted quantum mechanical calculations, using the density functional theory (DFT) code VASP [6]. VASP is a highly parallelized plane wave code, which describes the electron-ion interaction via pseudopotentials and uses the projector-augmented wave (PAW) method [7]. Exchange and correlation were accounted for by the general gradient approximation as introduced by Perdew, Burke and Ernzerhof [8]. To guarantee for sufficient accuracy, spin-polarized calculations with an energy cutoff of 600 eV were conducted using a $5 \times 5 \times 5$ k–point mesh for the $2 \times 2 \times 2$ CeO_2 supercell with 64 atoms and its doped derivatives. To investigate the impact of alkaline metal insertion, different numbers of alkaline metal atoms were added to the structure. For the relaxed structures, the atomic arrangements of alkaline metals and dopants were investigated and the charge distribution was analyzed by Bader analysis as well as by investigating the localized magnetic moments on the dopant species. For the above described structural optimization of the unit cells with different doping elements (Fe, Co) and the different alkaline metal concentrations (for Li, Na and K) atoms, typically two nodes with 20 cores each were used. The convergence of the electronic steps turned out to be rather slow, which is essentially due to Cerium. Similarly, for Bader analysis two 20 core nodes were used.

3 Results and Discussion

The cubic structure of CeO_2 and Fe-doped CeO_2 is shown in Fig. 1. The substitution of a Ce by an Fe atom results in an off-centering of the Fe atom. While in the pristine structure it is located in the center of an oxygen cube, the DFT calculations show a shift of the Fe atom towards a cube face. This can be understood in terms of the Ce–O distances inside this cube. They are significantly larger as the typical Fe–O distances in Fe–oxides, such that the Fe prefers to move toward the cube faces to decrease some Fe–O distances. This off–centering moreover results in more space in the structure that may be one of the reasons for the increased capacity. It has to be noted that so far only doping by one Fe atom was considered, while the arrangement of Fe atoms with respect to each other may further enhance this effect.

Apart from the off-centering a further, even more surprising effect is observed. First, the insertion of alkaline metal atoms in the direct vicinity of the dopant atom is

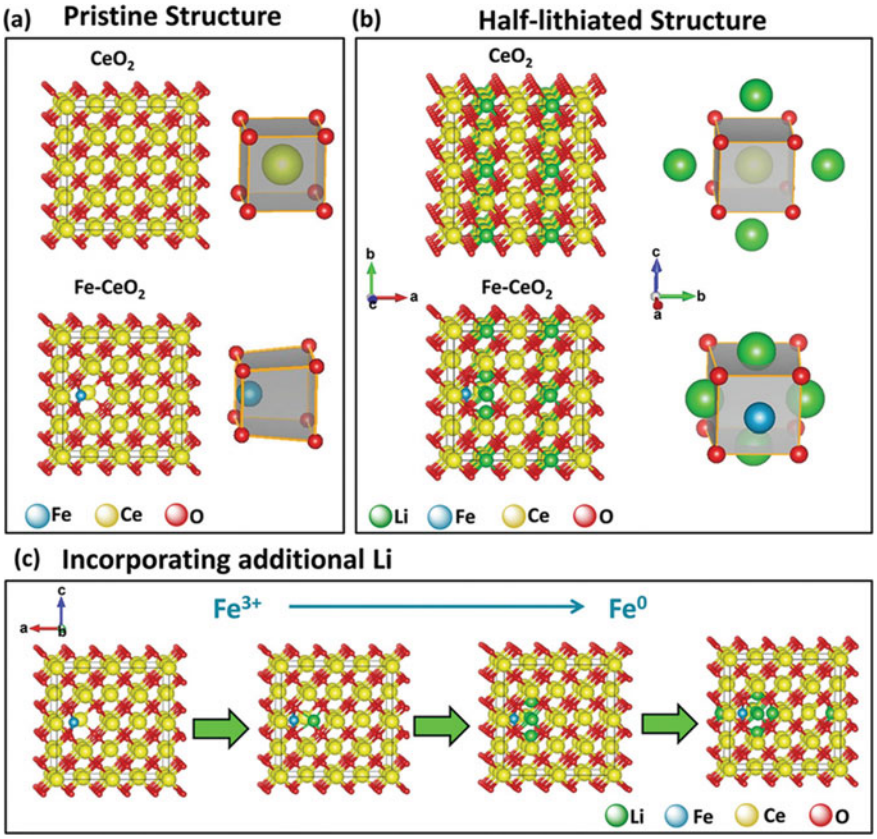


Fig. 1 The crystal structure of pristine and Fe-doped CeO_2, together with the local environment of the Fe/Ce site are shown in a), whereas the local environment after Li insertion is shown in b). In c) the successive insertion of Li and the reduction of Fe are schematically depicted. Figure reprinted from [9].

preferred and also results in adaption of the alkaline metal positions. They are shifted towards the Fe dopant as also depicted in Fig. 1b). Finally, an analysis of the charge distribution for increasing alkaline metal contents, as depicted in Fig. 1c), shows an unexpected behavior. Both, Bader analysis and investigation of the Fe magnetic moments, clearly indicate that the inserted alkaline metals first transfer their charge to the metal dopant. In fact, both measures point to the Fe dopant being reduced to its metallic state, i.e. Fe^{3+} is reduced to Fe^0. With this theoretical finding, the interpretation of experimental XANES data is corroborated, which indicated such an unexpected behavior. Indeed, the qualitative picture for the different alkaline metals is the same, such that in call cases the preferential reduction of Fe^{3+} seems to be the driving force for an increased capacity. Finally, the substitution of Fe by Co is

currently investigated. Here, the off–centering is also clearly observed, whereas the alkaline metal insertion is currently still under investigation.

4 Conclusion and Outlook

The study clearly shows that the increased capacity in Fe-doped CeO_2 is a consequence of the off-centering of the dopant. Moreover, a surprising reduction of Fe to its metallic state can be confirmed. Apart from the further investigation of the Co systems, other metal dopants will be investigated, to further optimize the capacity of CeO_2 based anodes. Moreover, an investigation of higher dopant concentrations is of great interest. In the above discussed results only isolated dopants were considered. The investigation of clustering of dopant atoms and the computational insertion limit are also of interest for further studies.

5 Introduction to the Second Part

Due to the high abundance and sustainability of zinc, aqueous Zn metal batteries are expected to reduce cost and environmental impact, thus making it a promising complementary technology to Li-ion batteries [10, 11]. Experimentally, the δ–$Ca_{0.24}V_2O_5$ derived V_2O_5 (CVO) shows a high capacity of more than 350 mAh/g, going along with a good capacity retention [12, 13]. However, the origin of this large capacity is strongly debated. In particular, the question arose which species are intercalated. Experimental data clearly indicate that both Zn^{2+} and H^+ can be stably intercalated between the layers. Moreover, an observed pH change during the discharge can be interpreted as a Zn^{2+}/H^+ exchange mechanism. The structure of CVO is complicated by structural water between the layers (see Fig. 2), such that standard methods are not able to identify the intercalated species. In particular, additional H^+ ions are not directly detected (only the just mentioned pH change in the aqueous solution indicated H^+ intercalation).

6 Computational Methods

To gain insights into the underlying mechanism, we have performed density functional theory (DFT) calculations. For this purpose, a large number of model structures with different intercalation chemistry were optimized by applying the VASP code [6]. Within the VASP code the electron-ion interaction is effectively described by the pseudopotential approach, which was applied in the generalized form of the projector augmented wave method [7]. For this work, spin polarized calculations were performed and the generalized gradient approximation in the formulation of Perdew,

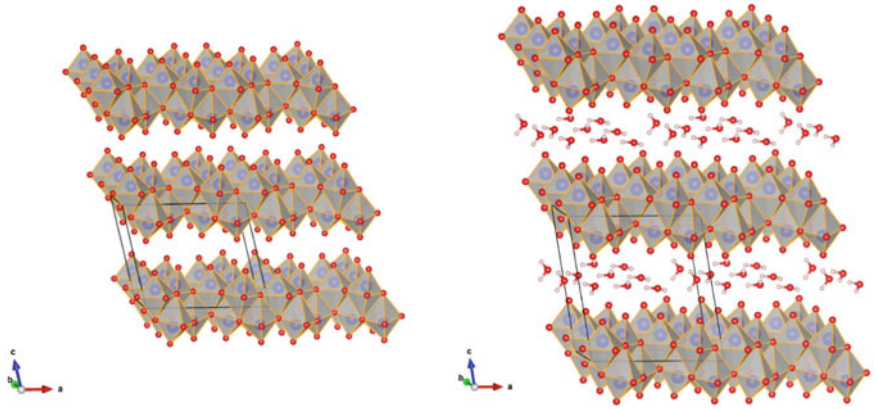


Fig. 2 The crystal structure of the investigated, layered V_2O_5 phase without (left) and with (right) structural water between the layers. Adapted with permission from [14]. Copyright (2020) American Chemical Society.

Burke and Ernzerhof (PBE) was applied to account for exchange and correlation [8]. Starting point for our computational study was the structure of the δ–$Ca_{0.24}V_2O_5$ phase. As a first step, a water-free V_2O_5 layer structure with 28 atoms in the unit cell was optimized with respect to lattice parameter and atomic positions, using an energy cutoff of 600 eV and a $3 \times 9 \times 3$ Γ–centered k–point mesh. In subsequent calculations different numbers of H and Zn atoms were inserted in between the layers of the structure, allowing for the investigation of the respective intercalation process. Furthermore, to check the impact of the structural water, the different structures were reinvestigated by adding four water molecules between the layers.

For the computational study, typically two nodes with 20 cores each were used for the geometry optimization. The complicated structure for the water containing case needed a large number of ionic states to converge to the energetic minimum (due to the rather shallow potential for the water molecules in between the layers).

7 Results and Discussion

In a first approach, we have neglected structural water only regarding the intercalation of H^+ and Zn^{2+} ions in CVO. Interestingly, it is observed that H^+ and Zn^{2+} prefer different positions in the layer and moreover a structural stability limit of 0.5 H^+/f.u. exists, whereas the intercalation of one Zn^{2+}/f.u. is still possible. Accessing the respective stability of intercalation compounds in an electrochemical environment, can be achieved by adapting the concept of the computational hydrogen electrode (CHE) [15, 16]. By this approach, the phase diagram with respect to the electrochemical potential of zinc and hydrogen ($\Delta\tilde{\mu}_{Zn}$ and $\Delta\tilde{\mu}_{H}$ respectively) can be constructed

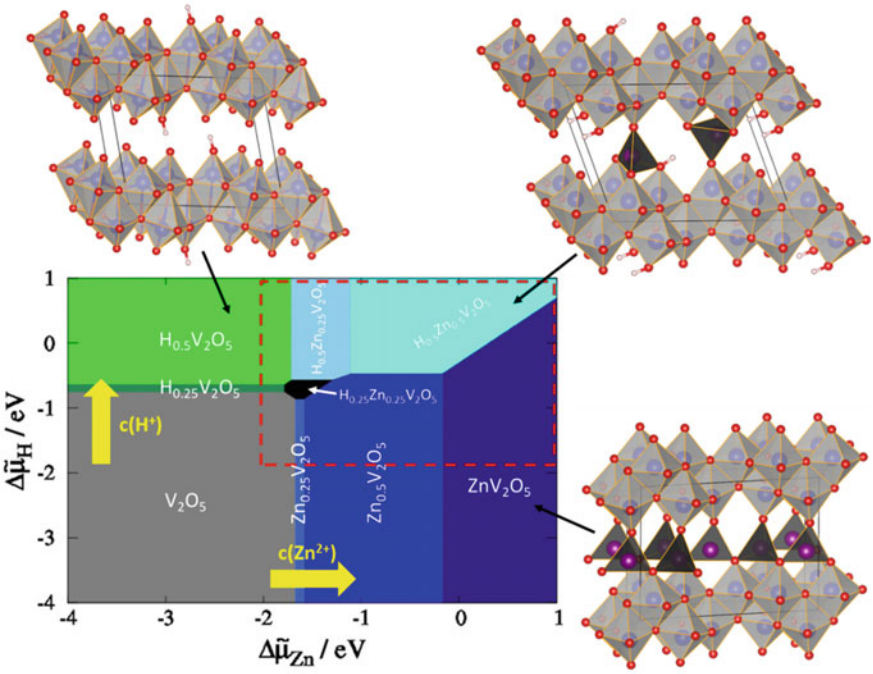


Fig. 3 Calculated phase diagram for hydrogen and Zn intercalation in water-free V_2O_5 as function of the respective chemical potential. The stoichiometries of the stable phases are denoted in the corresponding area of the phase diagram. Structure images, depicting hydrogen intercalation, Zn intercalation and hydrogen-Zn co-intercalation are exemplary shown next to the respective regions of the phase diagram. The dashed red box is a rough estimate of the chemical potential range relevant for the experiment. The yellow arrows point in the direction of increasing Zn and hydrogen concentration (i.e., decreasing pH). The areas marked by a star have been increased for better visibility. Adapted with permission from [14]. Copyright (2020) American Chemical Society.

from standard DFT calculations (i.e. without having to account for solvation energies). The phase diagram obtained with this approach is depicted in Fig. 3. Indeed, depending on the electrochemical potential, different intercalation compounds are stable. For understanding this phase diagram one has to keep in mind that in this graph, an increasing H^+ concentration (decreasing pH), corresponds to a displacement along the y–axis. This means that there are many regions in the phase diagram where H^+ will move out of the structure and Zn^{2+} moves in when the pH value increases. This corresponds exactly to the situation that is observed in experiment and hence strongly corroborates an exchange mechanism.

Next, we have increased the complexity of our model structures by including structural water. In the presence of structural water additional protons either form H_3O^+ ions with the structural water or adsorb on the V-O layers (this is what was also observed for the water-free case). Interestingly, the water containing structure can accept more than 1 H^+/f.u. Moreover, at high Zn contents, even a dissociation

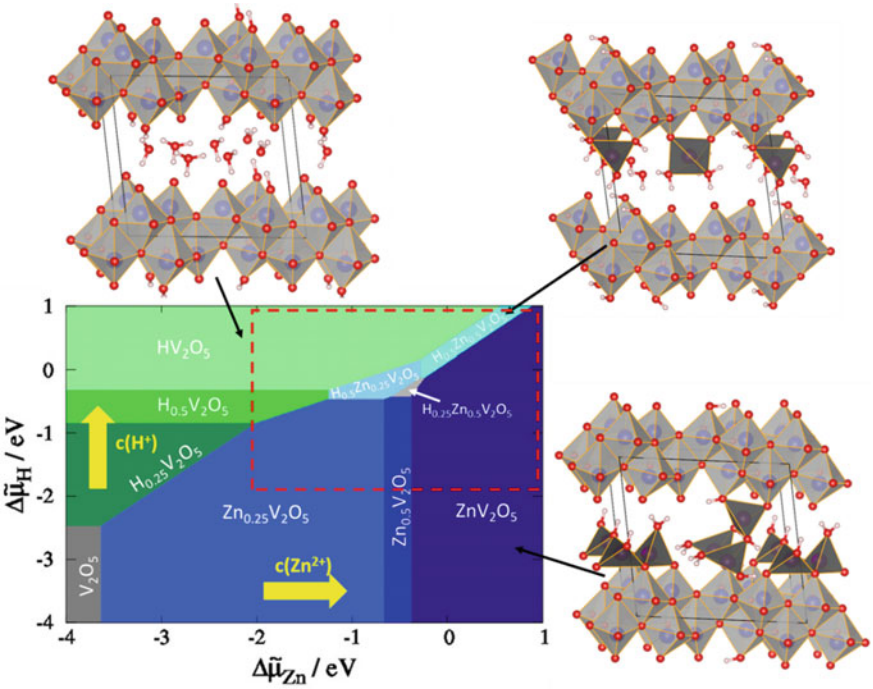


Fig. 4 Calculated phase diagram for hydrogen and Zn intercalation in water-containing V_2O_5 as function of the respective chemical potential (details are given in the caption of Fig. 3). Adapted with permission from [14]. Copyright (2020) American Chemical Society.

of structural water is observed in our model system, which is a consequence of the formation of O-H groups. For the phase diagram, the structural water results in changed phase fractions and potential ranges. However, the qualitative picture remains valid. Indeed, there are still several phase boundaries which will promote an exchange of H^+ and Zn^{2+} when they are crossed. While the details of the phase diagram will depend on the water content and the exact arrangement of the water molecules, the occurrence of phase boundaries that result in H^+/Zn^{2+} exchange are expected to be independent of the exact structural model, and therefore our findings strongly corroborate the proposed exchange mechanism Fig. 4.

8 Conclusion

A DFT based investigation of the Zn^{2+}/H^+ intercalation in CVO could nicely show that an exchange mechanism is possible. Applying the CHE concept allows to take the electrochemical environment into account and enables the determination of the phase diagram as a function of the electrochemical potential.

Acknowledgement The author wants to acknowledge discussion and co–operation with Yanjiao Ma, Yuan Ma, Gabriele Giuli, Giovanni Orazio Lepore, Francesco d'Acapito, Dorin Geiger, Johannes Biskupek, Ute Kaiser, Hanno M. Schütz, Anna Carlsson, Thomas Diemant, Rolf Jürgen Behm, Matthias Kuenzel, Xu Liu, Maider Zarrabeitia, Xinpei Gao, Guiseppe Antonio Elia, Stefano Passerini, Dominic Bresser and Axel Groß.

References

1. E.C. Evarts, Lithium batteries: To the limits of lithium. Nature **526**, 93 (2015)
2. J.B. Goodenough, How we made the Li-ion rechargeable battery. Nat. Electron. **1**, 204 (2018)
3. Y. Yamada, Y. Iriyama, T. Abe, Z. Ogumi, Kinetics of lithium ion transfer at the interface between graphite and liquid electrolytes: Effects of solvent and surface film. Langmuir **5**, 12766–12770 (2009)
4. N. Loeffler, D. Bresser, S. Passerini, M. Copley, Secondary lithium-ion battery anodes - From first commercial batteries to recent research activities. Johns. Matthey Technol. Rev. **59**, 34–44 (2015)
5. D. Bresser, K. Hosoi, D. Howell, H. Li, H. Zeisel, K. Amine, S. Passerini, Perspectives of automotive battery R&D in China, Germany, Japan, and the USA. J. Power Sources **382**, 176–178 (2018)
6. G. Kresse, J. Furthmüller, Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. Phys. Rev. B **54**, 11169–11186 (1996)
7. G. Kresse, D. Joubert, From ultrasoft pseudopotentials to the projector augmented-wave method. Phys. Rev. B **59**, 1758–1775 (1999)
8. J.P. Perdew, K. Burke, M. Ernzerhof, Generalized gradient approximation made simple. Phys. Rev. Lett. **77**, 3865–3868 (1996)
9. Y.-J. Ma, Y. Ma, G. Giuli, H. Euchner, A. Groß, G.O. Lepore, F. d'Acapito, D. Geiger, J. Biskupek, U. Kaiser, H.M. Schütz, A. Carlsson, T. Diemant, R.J. Behm, M. Kuenzel, S. Passerini, D. Bresser, Introducing highly redox-active atomic centers into insertion-type electrodes for lithium-ion batteries. Adv. Energy Mater. **10**, 2000783 (2020)
10. B. Tang, L. Shan, S. Liang, J. Zhou, Issues and opportunities facing aqueous zinc-ion batteries. Energy Environ. Sci. **12**, 11 (2019)
11. H. Qiu, X. Du, J. Zhao, Y. Wang, J. Ju, Z. Chen, Z. Hu, D. Yan, X. Zhou, G. Cui, Zinc anode-compatible in-situ solid electrolyte interphase via cation solvation modulation. Nat. Commun. **10**, 1 (2019)
12. G. Fang, J. Zhou, A. Pan, S. Liang, Recent advances in aqueous zinc-ion batteries. ACS Energy Lett. **3**(10), 2480–2501 (2018)
13. F. Wan, Z. Niu, Design strategies for vanadium-based aqueous zinc-ion batteries. Angew. Chemie - Int. Ed. **58**(46), 16358–16367 (2019)
14. X. Liu, H. Euchner, M. Zarrabeitia, X. Gao, G. Elia, A. Groß, S. Passerini, Operando pH measurements decipher H^+/Zn^{2+} intercalation chemistry in high-performance aqueous Zn/δ-V_2O_5 batteries. ACS Energy Lett. **5**(9), 2979–2986 (2020)
15. F. Gossenberger, T. Roman, A. Groß, Hydrogen and halide co-adsorption on Pt(111) in an electrochemical environment: A computational perspective. Electrochim. Acta **216**, 152–159 (2016)
16. J.K. Norskov, J. Rossmeisl, A. Logadottir, L. Lindqvist, J.R. Kitchin, T. Bligaard, H. Jonsson, Origin of the overpotential for oxygen reduction at a fuel-cell cathode. J. Phys. Chem. B **108**, 46 (2004)

Shear Induced Dynamic Grain-Refinement in Sliding Polycrystalline Metal Surfaces

Pedro A. Romero, Angelika Brink, Matthias Scherge, and Michael Moseler

Abstract Tribological shearing of conventional polycrystalline metal surfaces typically leads to grain refinement near the sliding interface, however, how and why grain refinement occurs in metallic surfaces under tribological shear remains poorly understood. Here, employing large scale atomistic simulations (∼5 to ∼44 million bcc iron atoms on 100 to 500 HPC cores) and ultra high vacuum microtribometry, we capture the mechanisms leading to the formation and evolution of the refined-grain layer and discuss its implications on the ongoing sliding motion. For pure bcc iron, the simulations showed that fundamentally the initial grains are refined through the generation of numerous dislocations and twin boundaries that quickly agglomerate and start forming non-crystalline walls, which eventually leads to the formation of relatively stable grain boundaries. As in the simulations, experimentally sheared pure iron surfaces showed significant grain refinement in the near surface region within the first two reciprocating sliding cycles with only minor changes in grain structure and very slow growth of the refined layer into the rest of the substrate for increasing number of sliding cycles. The simulations, more importantly, revealed that the evolution of the refined-grain structure is a dynamic continuously evolving process where after the initial grain structure is refined and a nanocrystalline layer is established, the refined grains are then continuously coarsened and refined by creating and moving grain boundaries, twin boundaries and other lattice defects. The motion of the grain boundaries is the main shear accommodating mechanism in the refined layer as the sliding motion continues. In essence, the generated refined layer hardens the surface and minimizes plastic flow towards the surface. Simultaneously, the dynamic coarsening and refining of the grains in the refined layer via grain boundary migration, allows the near surface material to undergo the necessary plastic shear deformation to be able to accommodate the imposed sliding motion by the counter surface.

P. A. Romero (✉) · A. Brink · M. Scherge · M. Moseler
Fraunhofer Institute for Mechanics of Materials IWM, Wöhlerstraße 11, 79108 Freiburg, Germany
e-mail: pedro.romero@iwm-extern.fraunhofer.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_12

1 Introduction

Despite their important technological implications, the underlying microscopic mechanisms controlling tribological processes in metallic polycrystalline solids remain widely unexplained [1–6]. Consequently, one has to frequently resort to empirical relations such as Amontons' friction law [7] (including its later refinement [8]) and Archard's wear law [9] in order to make engineering decisions about frictional resistance and wear. A widely observed phenomenon is that when conventional metallic surfaces are tribologically sheared, grain refinement [10–12] occurs near the sliding interface. Keeping the Hall-Petch strengthening curve in mind (i.e. increase in shear strength with decreasing grain size), it might seem counterintuitive, that metallic surfaces undergo grain refinement in order to decrease frictional resistance. Nevertheless, experimental *post-mortem ex-situ* examination of polycrystalline metallic substrates repeatedly shows that tribological shear leads to grain refinement near the sliding interface [10, 13, 14].

Unfortunately, the *in-situ* evolution and monitoring of the grain structure during tribological shearing of metallic surfaces remains experimentally inaccessible. Consequently, the underlying mechanisms leading to tribology induced grain refinement in metals is still unclear. It is not clear why and how exactly the originally microcrystalline near surface grains are transformed into a refined-grain (often nanocrystalline) layer near the sliding interface. Additionally, it is not clear how the generated refined layer facilitates the sliding motion. On the one hand, it is known that fine-grained nanocrystalline metallic surfaces exhibit in general higher hardness levels but lower ductility levels relative to conventional microcrystalline metals. On the other hand, it is expected that a highly deformable layer of material has to exist between two sliding counter bodies [15, 16]. Furthermore, due to the enhanced hardness of fine-grained nanocrystalline [17–19] metals, there is interest in using fine-grained nanocrystalline surfaces in order to reduce wear during tribological applications. However, it is not clear if nanocrystalline surfaces can accommodate severe plastic shear since they are known to be less ductile [20] than conventional polycrystalline metals. There is also no clarity on the stability of a fine-grained nanocrystalline surface during sliding and consequently there is no certainty on whether the nanocrystalline layer will recrystallize and/or undergo thermal [21] and/or mechanical [22] grain growth or grain refinement.

Metal surfaces are typically rough with contact only occurring at specific asperities. During sliding of metallic surfaces, it is at these contacting asperities where plastic deformation is mainly concentrated. The size of some of these asperity contacts can be small enough (tens or hundreds of nanometers) that they could be studied with large scale classical atomistic simulations, (i.e. molecular dynamics). Such simulations would allow direct *in-situ* observation of the generation and evolution of the refined-grain layer during tribological shearing. For polycrystalline metals, such atomistic simulations can provide a fundamental understanding of the mechanism responsible for the emergence and evolution of the observed grain refinement in clean ultra high vacuum (UHV) microtribometry experiments using equivalent high purity

metallic substrates. Here, motivated by the widespread use of iron (and its related alloys) in tribological applications, we have performed large scale molecular dynamics (MD) tribology simulations (∼5 to ∼44 million atoms on 100 to 500 HPC cores) and UHV microtribometry tribology experiments using pure bcc iron substrates as a model material in an effort to improve our understanding of the generation, evolution and consequences of the refined-grain layer near the sliding interface during tribological shearing.

2 Methodology

2.1 Experimental Set-Up

Microscale sliding experiments in ultrahigh-vacuum (UHV) conditions were performed using a spherical indenter on a flat substrate setup. The designed microtribometer, which is described in Ref. [23], bases the force measurements on the principle of the double leaf cantilever [24]. The substrate consisted of a 200 μm deep galvanic layer of pure iron bonded to an aluminum solid base. Before the sliding tests were performed, the iron surface was polished and sputtered until XPS of the surface exhibited a pure iron composition without any pollution (i.e. detectable iron-oxides). Figure 1(a) shows the initial microstructure after polishing and sputtering. The polishing procedure has no influence on the grain size, however sputtering causes a specific and reproducible roughness. The RMS roughness of the sample after sputtering was 3 nm on a scanning area of 5 μm^2. The indenter counter body was a ruby sphere with a 500 μm radius and a RMS roughness of 11 nm on a scanning area of 5 μm^2. Roughness measurements were performed with an Atomic Force Microscope. The sphere was cleaned with aceton and isopropanol separately before the sliding tests. During the entire process, the sample remained in high vacuum as it was transferred from the XPS chamber to the UHV tribometer via a manipulator. Only the transfer of the sample to the focused ion beam (FIB) (Zeiss-XB 1540) Microscope required exposure to environmental air. Different sliding friction experiments were performed at a normal load of 16 mN and a velocity of 8.3 μm/s or 100 μm/s for a sliding distance of 100 μm in each scan direction. The different experiments consisted of either 1, 10, 100 and 1000 reciprocating cycles on different wear tracks. In order to expose the crystal mircrostructural changes under the wear track, FIB cuts were performed through the wear track in the sliding direction after 1, 10, 100, 1000 cycles and for the initial substrate. Additionally, to detect the smaller grains sizes (∼30 nm), transmission electron microscropy (TEM) was performed for the 1000 cycle sliding test.

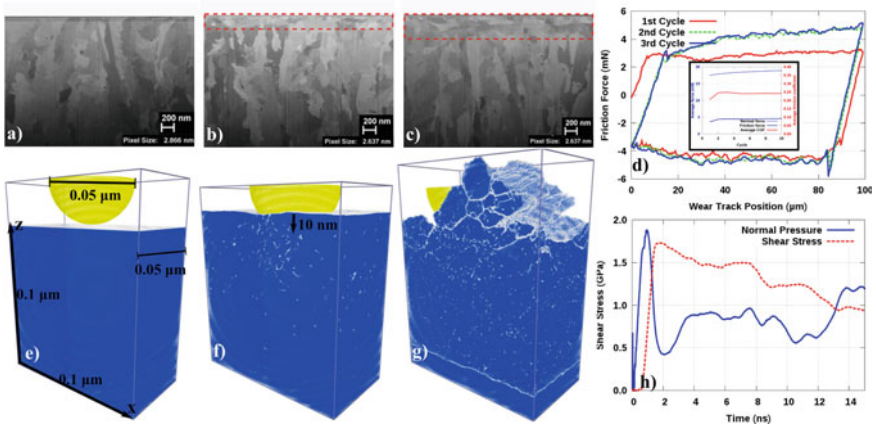


Fig. 1 Experimental and numerical observation of grain refinement near the surface of high purity iron substrates after being sheared with a hard spherical indenter. The initial grain structure of the microcrystalline iron substrate used in the experiments is shown in (a). The starting grain structure is significantly refined after just one sliding cycle as shown in (b). Further sliding up to 10, 100 and 1000 cycles resulted in slower refinement deeper into the substrate as show in (c). The sliding cycles were performed at F_n = 16 mN at either v = 8.3 μm/s or v = 100 μm/s. The average friction force after every cycle for the first 10 cycles is shown in (d). Relative to the first cycle, the friction force only shows small changes after the first cycle. A similar numerical setup using a pure iron single crystal substrate (100 nm × 51.5 nm × 100 nm) (~44 million atoms) along with a rigid spherical indenter exhibited a similar surface grain refinement. The initial numerical set-up is depicted in (e). The 3D monocrystalline substrate was first indented to a depth of ~10 nm at 20 m/s as shown in (f). Subsequently the 3D monocrystal is sheared by sliding the rigid indenter at 20 m/s over the indented surface at constant height. This simulated shear deformation results in grain refinement near the sliding interface as shown in (g) after ~15.1 ns (~300 nm) of sliding. The evolution of the average shear and normal stress for the simulated time is shown in (h).

2.2 *Numerical Models*

Massive molecular dynamics [25] simulations were carried out on three different atomistic models while employing an embedded atom method (EAM) interatomic potential for iron [26], which reliably models plastic deformation in pure iron for the applied loading and boundary conditions [27]. First a system composed of a massive 3D defect-free single crystal substrate along with a rigid 3D spherical indenter/slider was used to demonstrate that grain refinement can occur in 3D defect-free single grains. Second a model composed of a large scale quasi 3D bi-crystalline substrate along with a rigid cylindrical indenter/slider was used to reproduce the refined layer observed in FIB cuts through the length of the wear track produced by sliding experiments on pure polycrystalline iron surfaces. Lastly, a Lees-Edwards like uniform shear model was employed to demonstrate the dynamic equilibrium of the grain size during the refinement process.

A typical experimental set-up for tribology experiments is a ball on a substrate set-up with back and forth repetitive sliding. To mimic the experimental set-up, a 3D

atomistic model was constructed with a $1000 \times 515 \times 1000$ Å^3 (~44 million atoms) bcc iron monocrystalline substrate and a rigid spherical indenter with a 515 Å diameter as depicted in Fig. 1(e). The constructed monocrystalline substrate is first relaxed to 300 K for ~8 ps with periodic boundary conditions along the x- and y-direction with a fixed boundary at the bottom and a free surface at the top along the z-axis. The rigid indenter is then displaced ~10 nm into the substrate at 20 m/s. Subsequently the 3D single crystal substrate is sheared by sliding the rigid spherical indenter at 20 m/s over the indented surface at constant height and temperature (300 K) for ~15.1 ns (~3000 Å). For the stages of the simulation a 300 K dissipative particle dynamics (DPD) [28] thermostat with a dissipation constant of 0.05 eV ps Å^{-2} and a cutoff of 5.0 Å was applied to all non-rigid atoms in the system in order to reflect ambient temperature. Additionally, there is adhesion between the substrate and the rigid indenter.

To capture the evolution of grain structure directly underneath the center of the indenter as observed in FIB cuts under the wear track, an atomistic set-up was created with a bi-crystalline bcc iron substrate and a rigid cylindrical indenter as depicted in Fig. 2(a). The bi-crystalline substrate has x, y, z dimensions of $2000 \times 40 \times 1000$ Å^3 and can therefore be considered to be a quasi three-dimensional substrate. The initially fully periodic crystalline substrate contains two 0.1 μm size grains with a 45^o difference in lattice orientation and totaling ~5 million atoms. The constructed substrate is relaxed as a fully periodic system by minimizing with a conjugate gradient minimizer, then heating and holding at 500 K for 76 ps, then quenching from 500 to 300 K within 76 ps and finally holding at 300 K for 38 ps. All temperature changes during the annealing process are performed with a Berendsen thermostat [29] while relaxing the pressure in all directions with a Berendsen barostat [29]. This procedure resulted in a system with relaxed grains and grain boundaries. This quasi three-dimensional substrate is relaxed and held at 300 K during indenting and shearing. As in the massive model in Fig. 1(e), there is adhesion between the indenter and the substrate. The substrate is indented to a depth of ~8 nm prior to shearing at constant height. Both the indentation and scratching are performed at 20 m/s. Finally, in order to reduce thermal effects during sliding, the non-rigid atoms in the substrate are held at 300 K using a DPD [28] thermostat with a dissipation constant of 0.05 eV ps Å^{-2} and a cutoff of 5.0 Å.

Lastly, in order to investigate the stability of the generated refined grain structure, a uniform shear model was created using Lees-Edwards [30] like boundary conditions, which allows for the application of uniform shear deformation along the height of the specimen while maintaining 3D periodic boundary conditions. This model corresponds to a controlled volume within a micron-size grain directly underneath the sliding interface where the grains are expected to undergo uniform shear during sliding. As shown in Fig. 6(a), the simulations were conducted on a quasi three-dimensional $1000 \times 30 \times 1000$ Å^3 (~2.5 million atom) monocrystalline block of bcc iron in order to investigate the dynamic evolution of the refined grain structure. The initial 0.1 μm sized monocrystalline box is uniformly sheared at a strain rate of 9.969096e-4 s^{-1} (or 100 m/s) at a constant temperature of 300 K using a DPD [28]

thermostat in order to minimize thermal effects and to expose the effects of shear deformation on the grain structure evolution.

3 Results

3.1 Experiments

Post-mortem examination of FIB cuts along the wear track after different numbers of sliding cycles in the UHV micortribometer clearly reveals refinement of the grain structure underneath the sheared iron surface. As a reference, the FIB cut image in Fig. 1(a) presents the initial grain microstructure of the pure iron substrate used in the UHV microtribometry experiments. Initially the grains contain sizes on the order of micrometers in the vertical direction and on the order of 300 nm in the horizontal direction. The FIB cut image in Fig. 1(b) presents the *post-mortem* grain structure underneath the wear track after the first cycle of reciprocating sliding at 8.3 μm/s. Comparison of Fig. 1(a) with Fig. 1(b) clearly shows refinement of the initial grain structure underneath the sliding interface. The subsurface microstructure in Fig. 1(b), after just one sliding cycle, shows significant refinement in the subsurface grain structure down to a depth of ∼200 nm with the grains in this zone exhibiting a refined elongated shape along the sliding direction and ranging in size from 50 nm to 200 nm. The degree of grain refinement after the second cycle of reciprocating sliding is less drastic, nevertheless the refined layer continues to grow into the substrate with increasing number of cycles up to 10, 100 or 1000 cycles. The FIB cut image in Fig. 1(c) shows a clearly thicker refined layer (down to approximately 300 nm) for a sliding test up to 1000 sliding cycles at 8.3 μm/s. TEM images after 1000 sliding cycles reveal that the smallest grains in the upper 50 nm of the refined layer in Figs. 1(c) contain sizes on the order of 30 nm.

Figure 1(d) presents the evolution of the instantaneous lateral force during the first 3 sliding cycles for the sliding test at 8.3 μm/s. The main changes in the friction force occur during the first cycles. The friction force quickly increases during the first cycle and almost reaches its eventual steady state friction force value at the end of the retrace part of the first cycle. The drastic changes in friction force during the first trace of sliding can be related to surface plowing (i.e. flattening of initial roughness) and subsurface microstructural changes (i.e. refinement of the initial grain structure as discussed above). The inset in Fig. 1(d) presents the average friction and normal force for the first 10 sliding cycles where the average friction force per cycle was evaluated as the average lateral force during steady state sliding (i.e. between positions 30 to 70 μm on the wear track).

3.2 Simulations

Experiments similar to the ones presented above have repeatedly shown that tribological shearing of conventional microcrystalline metals leads to grain refinement near the sliding interface. Figure 1(c) is the typical *ex-situ post-mortem* experimental finding where the initial grain structure underneath the sliding interface is refined to a smaller grain size after the initial cycles of sliding. However, images like the one in Fig. 1(c) leads to questions such as: How is the refined layer formed? Why is a refined grain structure formed during sliding shear? How does a refined grain structure help or hinder the sliding motion between the sliding partners? How does sliding shear occur in the presence of the refined grain layer? Here, we try to address some of these questions using large scale classical molecular dynamics simulations of pure bcc iron, which closely resembles the metal used in the experiments presented above.

First, to verify the validity of our material model and to see if grain refinement occurs in fully three-dimensional perfect iron single crystals, simulations were conducted using the fully 3D atomic model of a bcc iron perfect single crystal and a rigid spherical indenter as depicted in Fig. 1(e). This numerical model closely resembles the experimental set-up of a spherical hard indenter sliding over a pure iron flat substrate. After indenting the free surface of the monocrystalline substrate to a depth of ∼10 nm with the rigid spherical indenter as shown in Fig. 1(f), numerous dislocations and other lattice defects are generated which propagate down into the substrate. As the spherical indenter slides over the free surface, it imposes severe shear on top of the substrate, which generates numerous additional dislocations that propagate into the substrate. Eventually these dislocations entangle and agglomerate until they form grain boundary walls. As shown in Fig. 1(g), significant grain refinement within the sheared surface material is generated by the sliding motion of the rigid indenter. The evolution of the average normal and shear stress resistance displayed by the monocrystalline substrate is displayed in Fig. 1(h). As expected, the normal stress reaches its highest value of ∼1.9 GPa at ∼600 ps while the shear stress stays close to zero during indentation. As sliding begins, the shear stress quickly increases to ∼1.7 GPa as the substrate is pushed to plastic slip to accommodate the imposed sliding shear. The gradual drop in shear stress, which occurs after about 2 ns, is accompanied by an increasing degree of grain refinement and by a reduction of undeformed material in front of the indenter as material flows to and around the sides of the spherical indenter as the wear-track is carved out. In total, the indenter slides over the free surface for only ∼15.1 ns, which corresponds to a sliding distance of about 3020 Å. Already within this small time window, sliding shear induced grain refinement occurs in an initially perfect single crystal. This results indicates that even in the absence of initial grain boundaries or other lattice defects, grain refinement can still occur.

Now, that we have seen that grain refinement can occur even in the case of a perfect three dimensional initial single crystal, we will investigate the evolution of the refined grain structure directly underneath the wear track and the deformation accommodation capability of the refined layer using a quasi-3D atomistic model. This

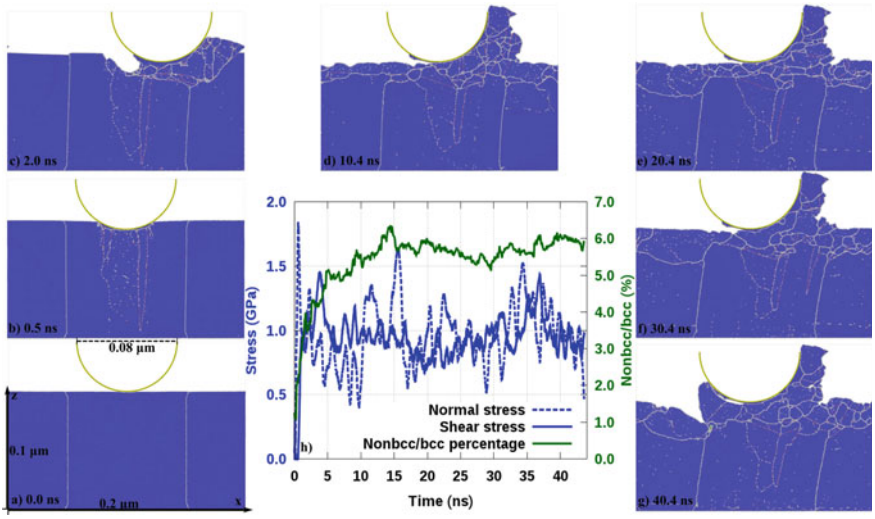


Fig. 2 Numerical atomistic observation of grain refinement on a pure iron quasi-3D bi-crystal with an initial grain size of 0.1 µm while indenting and shearing with a rigid cylindrical indenter (shown in yellow) at 20 m/s. The different snapshots show bcc atoms in blue, grain boundaries/lattice defects in white, and twin boundaries are depicted in red. The snapshot is (a) to (g) show the evolution of the initial grain structure from (a) the initial state to (b) the indented state to (c) the onset of shearing to the state of the grain structure after (d) one cycle, (e) two cycles, (f) three cycles, and (g) four cycles of sliding. The evolution of the normal stress, shear stress, and the non-bcc/bcc ratio percentage during the entire simulated indenting and sliding time is shown in figure (h).

model allows us to run the simulations for longer sliding times, which can show the growth of the refined layer into the initially unrefined part of the substrate. Using the model in Fig. 2(a), we have been able to capture the initial stages of grain refinement by indenting and scratching the top surface of the substrate. Figure 2(a) presents the initial relaxed bi-crystalline substrate along with a rigid cylindrical indenter. As shown in Fig. 2(b), indentation already causes the formation of a few initial grain boundary walls directly underneath the indenter. This initial boundaries contain high amounts of twin boundary atoms as indicated by the red colored region in Fig. 2(b). Scratching the surface of the substrate with the indenter, while holding the vertical position of the indenter constant, creates many more plastic dislocation defects and twin boundary atoms, which cause the grain structure of the substrate underneath the indenter to be refined as shown in Fig. 2(c). Continued sliding of the rigid cylindrical indenter up to one full cycle (0.4 µm) extends the initial grain refinement to the entire upper surface as shown in Fig. 2(d). As shown in Fig. 2(e), continued sliding of the indenter up to 2 sliding cycles (0.8 µm across the horizontal length of the substrate) only resulted in minor additional refinement into the substrate. Further sliding of the rigid indenter up to 3 (1.2 µm), and 4 (1.6 µm) cycles extends the refined layer a bit further down into the substrate as shown in Fig. 2(f) and Fig. 2(g) respectively. The refined grain structure at the top surface is partly formed as nucleated twin boundaries

propagate leaving amorphous grain boundaries behind. The grain boundaries of the generated grains are also created by plastic lattice defects which obstruct each other, entangle, and agglomerate to form walls with some of these walls eventually forming relatively stable grain boundary walls. Sometimes the more stable grain boundaries serve as nucleation sites for new grains.

Figure 2(h) presents the evolution of the average shear and normal stress resistance along with the nonbcc to bcc ratio percentage exhibited by the bi-crystalline iron substrate in Fig. 2(a) as it is indented and sheared. The normal stress reaches its highest value of ∼1.8 GPa at ∼500 ps while the shear stress stays close to zero during indentation. As scratching begins, the shear stress resistance quickly increases to ∼1.4 GPa as the substrate is pushed to plastic slip. After this initial increase the shear resistance drops to ∼0.8 GPa and then oscillates around ∼0.8 GPa while sometimes reaching values as low as ∼0.6 GPa and values as high as ∼1.2 GPa. The jumps and drops in shear stress are due to the fluctuating resistance to plastic slip exhibited by the substrate as the indenter marches forward. The fluctuations in plastic shear resistance are accompanied by fluctuations in the degree of grain refinement. Similar to the shear stress, the non-bcc to bcc ratio percentage increases significantly to around 6.2% after about 15 ns of sliding (between 1 and 2 sliding cycles), then the ratio fluctuates between 5% and 6% for the reminder of the simulated sliding time. Sometimes the decreases in shear resistance correspond with increases in the non-bcc to bcc ratio percentage. Notice that unlike the model in Fig. 1(e), the model in Fig. 2(a) does allow for a clearing of the wear track since the indenter is cylindrical and there are periodic boundary conditions along the thickness (y-direction) of the substrate. This means that the fluctuations in shear resistance are directly related to the fluctuations in degree of grain refinement underneath and in front of the indenter.

To expose the composition of the non-bcc (i.e. non-crystalline) atoms and to try to elucidate the formation of grain boundary walls, Fig. 3 presents the snapshots from Fig. 2 with all the bcc (i.e. crystalline) atoms removed. Figure 3(a) clearly shows that prior to indentation, the substrate was in a perfect bi-crystalline state. As shown in Fig. 3(b), the indentation process created grain boundary walls which extended almost the entire vertical dimension of the substrate. The form of these initial grain boundary walls comes from the type of plastic deformation imposed by the indenter. Indentation causes the substrate to deform mainly in the vertical direction and therefore causes plastic slip which propagates downwards. The scratching motion, on the other hand, pushes the material in the vicinity of the upper surface to deform and undergo plastic slip mainly in the horizontal direction. This is evident in Fig. 3(c)–(g), where the majority of grain boundaries are located near the scratched surface. Keep in mind that these surface grain boundaries are formed as the indenter nucleates plastic slip and twin boundaries which propagate and obstruct each other eventually forming the grain boundary walls seen in Fig. 3(g). Due to the concentration of plastic deformation near the sheared upper surface, there is a clear higher density of lattice defects within the refined layer. Figure 3(h) clearly depicts that while the fraction of twin boundary atoms remains relatively constant after the onset of sliding, the fraction of non-bcc atoms continues to increase during sliding due to the growing fraction of atoms in non-crystalline lattice positions up to ∼14 ns. After ∼14 ns, the total fraction of

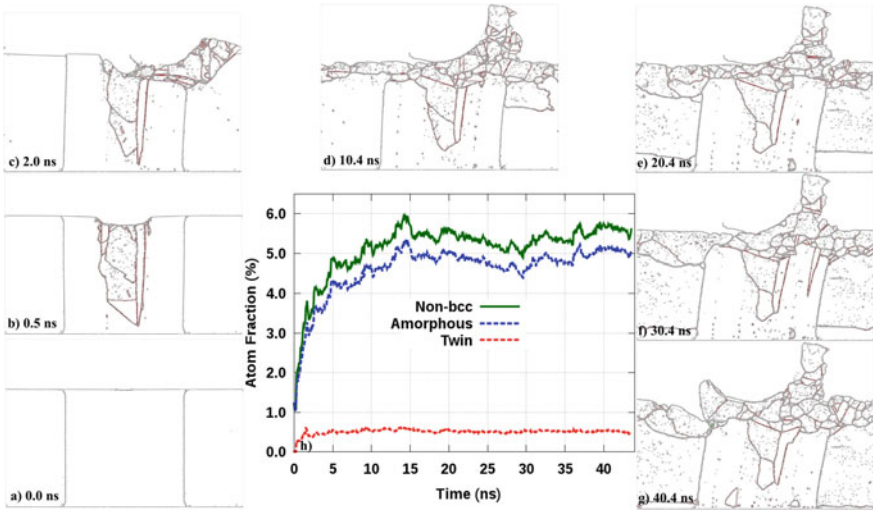


Fig. 3 Evolution of the non-crystalline atoms for the grain refinement case presented in Fig. 2. The snapshots in (a) to (g) show the evolution of non-crystalline atoms from (a) the initial state to (b) the indented state to (c) the onset of shearing to the state of the grain structure after (d) one cycle, (e) two cycles, (f) three cycles, and (g) four cycles of sliding. The evolution of the twin boundary atoms, the amorphous atoms, and the total (Non-bcc) of all twin boundary and amorphous non-crystalline atoms are shown in (h).

atoms in a non-crystalline state fluctuates between ∼4.9% and ∼5.8%. This indicates that the degree of surface grain refinement reaches a dynamic equilibrium within 1 to 2 sliding cycles as in the experiments. The fluctuations in the fraction of non-bcc atoms after ∼14 ns are due mainly due to grain growth and grain refinement directly underneath and in front of the indenter (see Fig. 3(e)–(g)).

As shown in Figs. 1 and 2, even though grain refinement can extend down into the substrate as sliding continuous, the most significantly refined layers remain close to the sliding interface. This phenomenon is shown in Fig. 4 where the quantification of the vertical variation of the crystalline and non-crystalline atoms for an example snapshot (at 17.2 ns) is depicted. The localization of the refined region near the sliding interface is already clearly visible in Fig. 4(b). Figure 4(a) displays the variation of bcc and non-bcc along the height of the specimen in Fig. 4(b), and the plots clearly confirm the localization of grain refinement near the upper sliding interface. Figure 4(c) further indicates how the twin boundary atoms and the atoms in amorphous (i.e. non-crystalline) states (which compose the total fraction of non-bcc atoms) are concentrated near the sliding contact interface.

After a large number of plastic deformation lattice defects are accumulated within large perfect grains, as shown in Figs. 2 and 3, the agglomeration of the these lattice defects can lead to grain refinement. However, once the near surface grains are refined, they can not accommodate as much plastic deformation as in their unrefined initial crystal structures. Nevertheless, these refined surface grains still need to accommo-

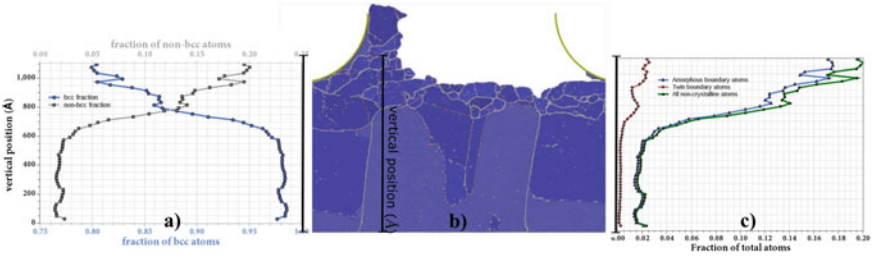


Fig. 4 Quantification of the localization of grain refinement near the sliding interface after more then one of cycle of sliding (17.02 ns) for the case presented in Fig. 2. (a) Variation of the fraction of bcc and non-bcc atoms along the height of the (b) sheared iron substrate after 17.02 ns. (c) The twin boundary atom and the amorphous atom contribution to the non-crystalline atom variation along the height of the substrate.

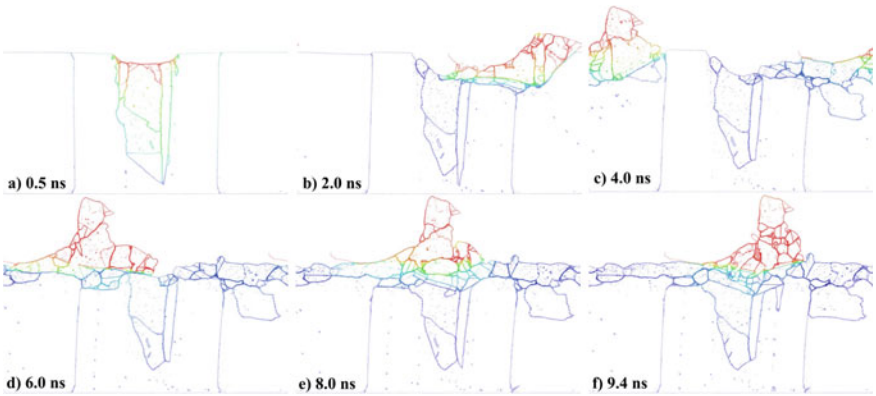


Fig. 5 Snapshots during the first sliding cycle indicating the grain boundary mobility during indenting and shearing for the bi-crystalline quasi-3D pure iron substrate presented in Fig. 2. Blue represents atoms with ∼0 m/s horizontal velocity and red represents atoms with a horizontal velocity of 20 m/s. All bcc crystalline atoms have a ∼0 m/s horizontal velocity and have been removed from every snapshot here. The snapshot in (a) shows that during indentation, the motion of the grain boundaries propagates deep into the substrate underneath the indenter. However, the snapshots in (b) to (f) show that during shearing, only the grain boundaries directly underneath and in front of the indenter exhibit high mobility. Grain boundaries far in front of the indenter on the surface do not exhibit mobility, which indicates that the shear deformation required to accommodate the sliding motion is generated by the migration and sliding of the grain boundaries directly underneath and in front of the indenter.

date extremely large amounts of plastic shear in order to accommodate the sliding motion of the indenter. To demonstrate how plastic deformation is accommodated within a refined surface grain structure, Fig. 5 shows the grain boundary mobility during the first cycle of sliding motion. Keep in mind that all crystalline atoms have zero relative mobility and have been removed to improve the visualization in Fig. 5. After indentation of the substrate as shown in Fig. 5(a), which show grain boundary mobility downwards into the substrate, the snapshots in (b) to (f) show that during

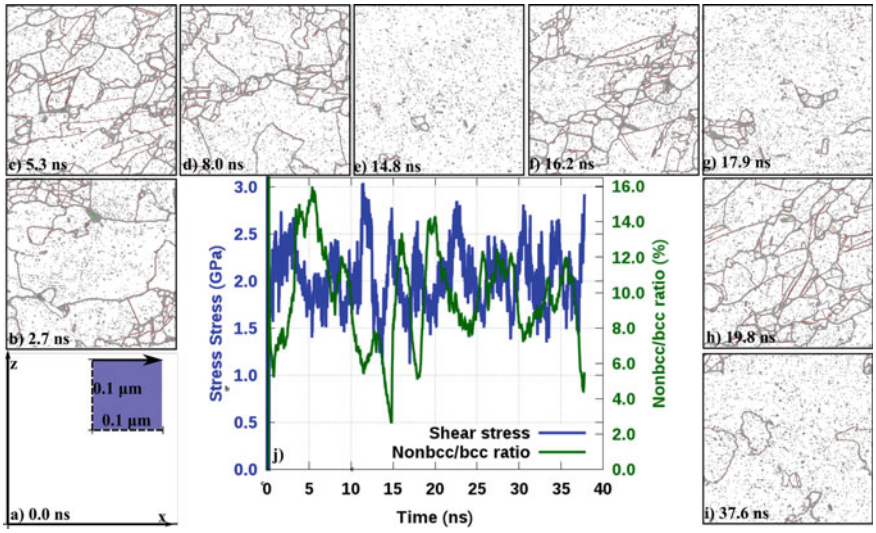


Fig. 6 Evolution of the crystal structure for a uniformly sheared 0.1 μm quasi-3D pure iron single crystal. Snapshot (a) shows the perfect single crystal structure at the beginning. The inset in (a) shows the completely bcc state of the sample at the beginning. Snapshots (b) to (i) show the fluctuating nature of the crystal structure (through the non-crystalline atoms) during continuous uniform shearing of the sample. Sometimes the sample can reach extremely refined states such as those in (c) and (h) or moderately refined states such as those in (d) and (f) or virtually single crystal states such as those in (e), (g) and (i). The quantification of the evolution of the shear stress and the ratio percentage of non-bcc to bcc atoms clearly shows the dynamic fluctuating nature of the grain refinement process.

shearing, only the grain boundaries directly underneath and in front of the indenter exhibit high mobility. Grain boundaries far in front of the indenter near and underneath the surface do not exhibit mobility. This indicates that the shear deformation required to accommodate the sliding motion is generated mainly by the migration of the grain boundaries directly underneath and in front of the indenter. Therefore grain boundary migration is the main source of shear deformation in refined metallic surfaces under tribological shear loads.

We also investigated grain refinement via uniform shear of a fully periodic quasi three-dimensional monocrystalline specimens with a 0.1 μm initial grain size. This approach allows us to investigate the dynamic equilibrium of grain refinement underneath the sliding interface within the bulk of an iron polycrystalline metal where the grains are expected to be uniformly sheared by the sliding motion. Figure 6 presents an example simulation where a perfect iron single crystal undergoes grain refinement by simply being uniformly sheared along the horizontal direction using Lees-Edwards like boundary conditions. The grain refinement process is the same as that described in Fig. 2 and Fig. 3, where the generated dislocations and twin boundaries interact until relatively stable grain boundary walls are formed. Interestingly, the different snapshots, from Fig. 6(a) to (i), reveal how the degree of grain refinement

fluctuates as the initial monocrystal is increasingly sheared up to ∼38 sliding cycles (i.e. ∼38 times the horizontal length of the specimen). The specimen can reach extremely refined states such as those in Fig. 6(c), (f) and (h), but the specimen can also acquire states with very low grain refinement such as those in Fig. 6(e), (g) and (i). Figure 5(f) presents the evolution of the nonbcc to bcc atom ratio along with the shear stress along the shear direction. The nonbcc to bcc atom ration clearly displays a fluctuating seesaw evolution around ∼10%, which demonstrates that grain refinement is a dynamic process where every time the grains are refined, they are subsequently coarsened and then refined again. Interestingly, the peaks and valley for the shear stress respectively correspond to the valleys and peaks of the nonbcc to bcc ratio percentage. This indicates that in general the surface of a material undergoing tribological shear exhibits a lower resistance to plastic shear in its most refined states because the mobility of the instantaneous grain boundaries is able to more easily accommodate the sliding shear relative to dislocation nucleation and motion. This perhaps counterintuitive point, can be understood if we recall that the grain boundaries formed directly underneath the surface undergoing shear can have high mobility (as depicted ins Fig. 5 for the bicrystalline model).

Finally, Fig. 7 clearly shows the fluctuating nature of the grain refinement process by counting the number of grains as the subtrate in Fig. 6 is uniformly sheared. This analysis also indicated that counting the number of grains in the systems is virtually equivalent to counting the ratio of non-bcc to bcc atoms in our bcc iron-

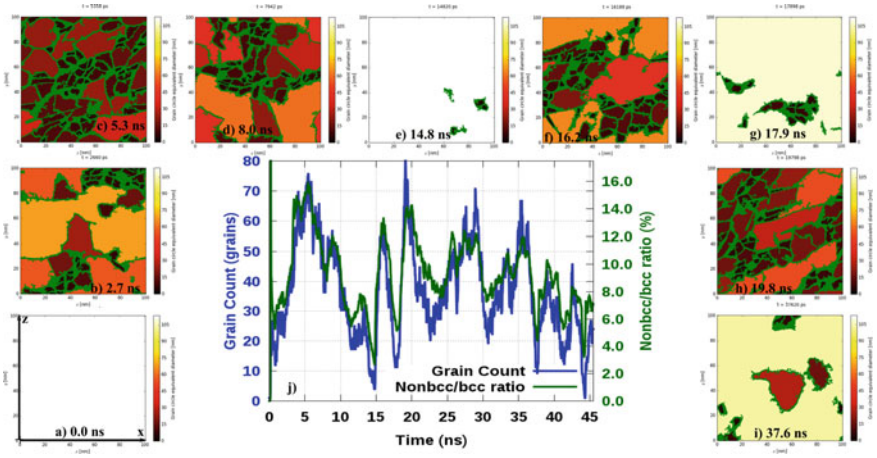


Fig. 7 Evolution of the grain size for the uniformly sheared iron single crystal in Fig. 6. Snapshot (a) show the perfect single grain at the beginning. Snapshots (b) to (i) show the fluctuating nature of the grain-size during continuous uniform shearing of the sample. Sometimes the sample can reach extremely refined states such as those in (c) and (h) or moderately refined states such as those in (d) and (f) or virtually single crystal states such as those in (e), (g) and (i). The quantification of the evolution grain count and the ratio percentage of non-bcc to bcc atoms clearly shows that both the non-bcc to bcc ration or the grain are good and identical indications of the dynamic fluctuating nature of the grain refinement process in sheared metals.

substrates. Figure 7 also clear shows that once a single crystal or a coarse grained system is initially refined through generation and agglomeration of lattice defects that are necessary to accommodate the imposed shear deformation, the newly refined grain structure will then be coarsened by the migration of grain boundaries, which is necessary to continue to accommodate the large amount of plastic shear deformation that is being imposed on the system. Once the refined grain structure is sufficiently coarsened, it will again be refined through the generation of new lattice defects. Therefore, as long as the system is continuously being sheared, it never reaches a steady state refined grain structure, but fluctuates continuously between highly refined and highly coarsened grain structures.

4 Conclusion

Post-mortem examination of tribologically sheared metallic surfaces typically reveals a refined nanocrystalline layer near the sliding surface. This article provides an atomistic observation and explanation for this commonly observed near-surface grain refinement phenomenon in sliding metallic surfaces. Using large scale atomistic simulations (∼5 to ∼44 on 100 to 500 HPC cores), we capture the process of grain refinement on initially crystalline high purity bcc iron substrates. The simulations show that the initial near-surface grain structure is refined through the generation of dislocations and twin boundaries, which agglomerate into boundary walls with some of these walls eventually forming stable grain boundaries. As frequently shown by the experiments, the simulations displayed a very slow growth of the refined layer into the substrate after the initial sliding cycles. Most importantly, the simulations showed that the generated near-surface grain refinement is a dynamic ongoing process where once a refined nanocrystalline layer is established, the grains in this layer are subsequently coarsened and then refined again. This grain refinement and coarsening process repeats itself as the sliding motion continues without reaching a steady-state grain size. We have to keep in mind that surfaces under tribological shear loads need to allow the counter surface to slide over with minimal material flow towards the surface while still generating the necessary plastic shear deformation to accommodate the sliding motion. Therefore, by reducing the near surface grain size, the surface becomes harder and minimizes material flow towards the surface. However, the near-surface region still needs to undergo a significant amount of shear plastic deformation in order to accommodate the imposed sliding motion. The necessary plastic shear is generated through the localized mobility (i.e. migration) of the grain boundaries and twin boundaries (and other lattice defects) directly underneath and in front of the moving counter surface. For this reason as grain boundaries form and migrate during shearing, the refined layer fluctuates in a dynamic fashion between a highly refined and a significantly coarsened layer as the sliding motion continues. Grain boundary migration in the refined layer is therefore the main mechanism for the accommodation of the imposed shear deformation by tribological shear loads on polycrystalline metal surfaces.

Acknowledgements We gratefully acknowledge the computing time granted under projects ATICSI and ASTC on the supercomputer ForHLRII funded by the Ministry of Science, Research and the Arts Baden-Württemberg and by the Federal Ministry of Education and Research. We also acknowledege Dr. Alexander Held for assistance with post-processing of the evolution of grain refinement.

References

1. J. Krim, Surf. Sci. **500** (2002)
2. L. Pastewka, S. Moser, M. Moseler, Tribol. Lett. **39** (2010)
3. P.A. Romero, G. Anciaux, A. Molinari, J.-F. Molinari, Model. Simul. Mater. Sci. Eng. **20** (2012)
4. P.A. Romero, G. Anciaux, A. Molinari, J.-F. Molinari, Comput. Mater. Sci. **72** (2013)
5. P.A Romero, T.T. Järvi, N. Beckmann, M. Mrovec, M. Moseler, Phys. Rev. Lett. **113** (2014)
6. N. Beckmann, P.A. Romero, D. Linsler, M. Dienwiebel, U. Stolz, M. Moseler, P. Gumbsch, Phys. Rev. Appl. **2** (2014)
7. G. Amontons, Mémoire de l'Académie Royale A (1699)
8. F.P. Bowden, D. Tabor, The Friction and Lubrication of Solids (1950)
9. J.F. Archard, J. Appl. Phys. **24** (1953)
10. A. Emge, S. Karthikeyan, D.A. Rigney, Wear **267** (2009)
11. P. Stoyanov, P.A. Romero, T.T. Järvi, L. Pastewka, M. Scherge, P. Stemmer, A. Fischer, M. Dienwiebel, M. Moseler, Tribol. Lett. **50** (2013)
12. A.P. Zhilyaev, S. Swaminathan, A.I. Pshenichnyuk, T.G. Langdon, T.R. McNelley, J. Mater. Sci. **48** (2013)
13. B. Yao, Z. Han, Y.S. Li, N.R. Tao, K. Lu, Wear **271** (2011)
14. A. Moshkovich, V. Perfilyev, L. Meshi, S. Samuha, S. Cohen, H. Cohen, A. Laikhtman, L. Rapoport, Tribol. Int. **46** (2012)
15. M. Godet, Wear **100** (1984)
16. Y. Berthier, M. Godet, M. Brendle, Tribol. Trans. **32** (1989)
17. T. Hanlon, A.H. Chokshi, M. Manoharan, S. Suresh, Int. J. Fatigue **27** (2005)
18. Z. Han, L. Lu, K. Lu, Tribol. Lett. **21** (2006)
19. Y.S. Zhang, Z. Han, K. Wang, K. Lu, Wear **260** (2006)
20. K.M. Youssef, R.O. Scattergood, K.L. Murty, J.A. Horton, C.C. Koch, Appl. Phys. Lett. **87** (2005)
21. E.A. Holm, S.M. Foiles, Science **328** (2010)
22. J.C.M. Li, C.M. James, Phys. Rev. Lett. **96** (2006)
23. D. Marchetto, R. Benzig, S. Korres, M. Dienwiebel, Tribol. - Mater. Sur. Inter. **6** (2012)
24. M. Scherge, S.S. Gorb, Biological micro- and nanotribology (2001). https://doi.org/10.1007/978-3-662-04431-5
25. S. Plimpton, J. Comput. Phys. **117** (1995)
26. G. Simonelli, R. Pasianot, E.J. Savino, Mat. Res. Soc. Symp. Proc. **291** (1993)
27. M. Müller, P. Erhart, K. Albe, J. Phys.: Conden. Matter **19** (2007)
28. R.D. Groot, P.B. Warren, J. Chem. Phys. **07** (1997)
29. H.J.C. Berendsen, J.P.M. Postma, W.F. van Gunsteren, A. DiNola, J.R. Haak, J. Chem. Phys. **81** (1984)
30. A.W. Lees, S.F. Edwards, J. Phys. C: Solid State Phys. **5** (1972)

Multi-physics Multi-scale HPC Simulations of Skeletal Muscles

Aaron Krämer, Benjamin Maier, Tobias Rau, Felix Huber, Thomas Klotz, Thomas Ertl, Dominik Göddeke, Miriam Mehl, Guido Reina, and Oliver Röhrle

Abstract We present a highly scalable framework for the simulation of skeletal muscles as a neuromuscular system. Our work is based on previous implementations of a complex model coupling different physical phenomena on different temporal and spatial scales, in particular bio-chemical processes on the cellular level (as ordinary differential equations), electrical signal propagation along muscle fibers (as many one-dimensional diffusion equations), and the EMG signal in the three-dimensional muscle at organ scale. Our contribution is a new software toolbox that allows to

A. Krämer (✉)
Institute of Applied Analysis and Numerical Simulation, University of Stuttgart, Stuttgart, Germany
e-mail: aaron.kraemer@ians.uni-stuttgart.de

B. Maier
Institute of Parallel and Distributed Systems, University of Stuttgart, Stuttgart, Germany

T. Rau · G. Reina
Visualization Research Center, University of Stuttgart, Stuttgart, Germany

F. Huber
Institute of Applied Analysis and Numerical Simulation, Institute of Parallel and Distributed Systems, University of Stuttgart, Stuttgart, Germany

T. Klotz
Institute for Modelling and Simulation of Biomechanical Systems, University of Stuttgart, Stuttgart, Germany

T. Ertl
Stuttgart Center for Simulation Science, Visualization Research Center, University of Stuttgart, Stuttgart, Germany

D. Göddeke
Institute of Applied Analysis and Numerical Simulation, Stuttgart Center for Simulation Science, University of Stuttgart, Stuttgart, Germany

M. Mehl
Institute of Parallel and Distributed Systems, Stuttgart Center for Simulation Science, University of Stuttgart, Stuttgart, Germany

O. Röhrle
Institute for Modelling and Simulation of Biomechanical Systems, Stuttgart Center for Simulation Science, University of Stuttgart, Stuttgart, Germany

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_13

generate simulation code suited for the execution on a supercomputer from the commonly used XML-based model-description used in the respective community. We present different variants of this code generation for CPUs, specific optimizations for our muscle model, and our in situ visualization approach that allows us to minimize data transfer between simulation, and the visualization front-end (such as the Powerwall at VISUS (https://www.visus.uni-stuttgart.de/)) along with results for numerical experiments on up to approximately seven thousand cores of the CRAY XC40 system Hazel Hen (https://www.hlrs.de/systems/cray-xc40-hazel-hen/) for more than 180,000 muscle fibers. The original implementation was limited to 4 cores. Compared to our previous work in [4], we further enhanced scalability, but in particular also node-level performance.

Keywords Neuromuscular system · Code generation · Equation coupling · Performance optimization · In situ visualization

1 Introduction

The human musculoskeletal system can support many different actions, from applying strong forces to very tactile movements. Even 'simple' tasks like grabbing an object involve highly coordinated actions, i.e., voluntary contractions of skeletal muscles. Understanding the underlying mechanism of neuromuscular control is challenging and subject to active research. Besides obtaining new insights into the basic understanding of the neuromuscular system itself, a detailed understanding paves the way to many highly beneficial applications, including but not limited to personalized rehabilitation, patient-specific prosthesis design, passenger safety in crash scenarios, surgical planning tools, human-computer interfaces, ergonomic assessment of the workplace and the design of assistive tools. The desired degree of detail and complexity within models for (parts of) the neuro-musculoskeletal system requires the coupling of different physical phenomena on different temporal and spatial scales, e.g., models describing the mechanical or electrical state of the muscle tissue on the organ scale and the bio-chemical processes on the cellular scale (cf. Sect. 2.1). One of the few non-invasive and clinically available diagnostic tools to obtain insights into the functioning (or disfunctioning) of the neuromuscular system are electromyographic (EMG) recordings. They measure the activation-induced potentials on the skin surface, and rely on the direct connection between the discharge of spinal motorneurons and the resulting action potentials of the associated muscle fibers. Due to the high complexity of the neuromuscular system, conclusions from the EMG signals to the behavior of the motorneurons are notoriously difficult, in particular, as methods to decode the activation mechanisms cannot be fully validated based on experimental data.

Using synthetically generated simulation data can provide important contributions to the analysis of EMG signals. However, being able to take into account all these different processes on different scales requires a flexible multi-scale, multi-physics

computational framework and significant compute power. Over the past years, we have worked towards this goal, see [4] for intermediate results that we update here. We emphasize that only the joint development of model improvements, changes to the mathematical algorithms, suitable HPC techniques and, often forgotten, appropriate visualization techniques to generate a feedback loop with the domain scientist can lead to true progress.

In this chapter, we describe and analyze the HPC-related improvements we achieved, both in terms of runtime and scalability, but also in terms of an in situ visualization on Hazel Hen, the Tier-0 system installed at the HLRS in Stuttgart, that peaked at position #8 in the TOP500 in November 2015. In particular, we demonstrate techniques that enable the efficient simulation of muscles comprising a realistic number of 100,000s of muscle fibers. The remainder of this chapter is organized as follows: In Sect. 2 we present the multi-scale model and the associated numerical schemes. Section 3 covers the implementation and the optimizations we framed. Results are discussed in Sect. 4, and we conclude with a summary of the most important findings and an outlook to future work in Sect. 5.

2 Model and Discretization

A skeletal muscle consists of millions of sarcomeres which are arranged in series as a myofibril. Many myofibrils arranged in parallel then form a muscle fiber. About 10.000 to 1.000.000 muscle fibers form the active tissue of a muscle, depending on the type of muscle. Muscle fibers are much longer than wide. Also, action potentials which initiate the force generation in selected fibers only spread along these fibers, not transversely. Thus, all sarcomeres across a muscle fiber behave similarly, and we can represent all sarcomeres of such a muscle fibers' cross section as a zero-dimensional *subcell*. This is a common procedure in skeletal muscle modelling. In addition to a diffusive spreading of the action potential, there is a reactive subcellular behavior which can have both a supporting and a suppressive effect on the action potential propagation.

2.1 Model Formulation

Our muscle model describes the evolution of intra-cellular bio-chemical compositions, the propagation of action potentials and EMG signals. Action potentials are treated by modelling the physical quantity of transmembrane potential. A detailed description of the underlying bio-physical view on this subject is given in [15]. The interplay of the three components—bio-chemical compositions $\boldsymbol{y}$, transmembrane potentials V_m, and extracellular potential ϕ_e—is governed by a multi-scale coupling of the following differential equations, each on different domains:

$$\frac{\partial \boldsymbol{y}}{\partial t} = G\left(\boldsymbol{y}, V_{\mathrm{m}}, I_{\mathrm{stim}}\right) \qquad \text{on } \Omega_s^i, \quad (1)$$

$$\frac{\partial V_{\mathrm{m}}}{\partial t} = \frac{1}{AC_{\mathrm{m}}}\left(\frac{\partial}{\partial x}\left(\sigma_{\mathrm{eff}}\frac{\partial V_{\mathrm{m}}}{\partial x}\right) - AI_{\mathrm{ion}}\left(\boldsymbol{y}, V_{\mathrm{m}}, I_{\mathrm{stim}}\right)\right) \qquad \text{on } \Omega_f^j, \quad (2)$$

$$0 = \mathrm{div}\Big(\left(\boldsymbol{\sigma_e} + \boldsymbol{\sigma_i}\right)\mathrm{grad}\left(\phi_e\right)\Big) + \mathrm{div}\Big(\boldsymbol{\sigma_i}\,\mathrm{grad}\left(V_{\mathrm{m}}\right)\Big) \qquad \text{on } \Omega_m. \quad (3)$$

Here, Ω_s^i, $i = 1, \ldots, N_s$ are locations of N_s zero-dimensional subcells. Ω_f^j with $j = 1, \ldots, N_f$ are the one-dimensional representations of muscle fibers, and Ω_m is the three-dimensional muscle domain. It holds $\Omega_s^i \subsetneq \Omega_f^j$ for any subcell belonging to muscle fiber j, and $\Omega_f^j \subsetneq \Omega_m$ for any muscle fiber.

Equation (1) was originally formulated by Hodgkin and Huxley [12]. We refer to their article for a more detailed presentation of the right-hand side G. G is a nonlinear function, depending on the subcellular state $\boldsymbol{y}$, on the muscle fibers' transmembrane potential V_{m}, and the stimulation current I_{stim} stemming from one of many motor units of the central nervous system. Equation (1) forms a closing condition for the *Monodomain Equation* (2), which describes the evolution of the transmembrane potential V_{m} along a muscle fiber. The muscle fibers' surface to volume ratio is called A, C_{m} is the capacitance of its membrane, σ_{eff} its effective conductivity. The ionic current passing across the membrane is described by I_{ion}. By Eq. (3) a description of the extracellular potential ϕ_e is given throughout the three-dimensional tissue. We refer to it as *stationary Bidomain Equation*. The conductivity tensors in the extra- and intra-cellular domains are denoted as $\boldsymbol{\sigma_e}$ and $\boldsymbol{\sigma_i}$. The solution variable of Eq. (3), ϕ_e, corresponds to EMG signals over time at any spatial point of interest.

2.2 Discretization and Solution

The discretization and solution of Eqs. (1)–(3) follows the method described in [4] and will be summarized briefly in the following.

We discretize the spatial operators of the Monodomain Equations (2), by one-dimensional (1D) Finite Elements with linear ansatz functions, yielding 1D muscle fiber meshes. We use three-dimensional Finite Elements with linear ansatz functions to discretize the stationary Bidomain Equation (3) in the muscle volume Ω_m. Multiple 1D muscle fiber meshes are embedded in the 3D domain of the muscle volume. The value of the transmembrane potential V_{m}, is linearly interpolated and mapped between the 1D and the 3D meshes discretizing Ω_f^j and Ω_m.

The time discretization is visualized in Fig. 1. We apply a Strang operator splitting with time step $dt_{\mathrm{splitting}}$ to the Monodomain system, Eqs. (1) and (2), and, thus, solve the diffusion (1D) and reaction (0D) terms alternatingly. For the reaction term, we use (multiple) time steps of Heun's method with time step $dt_{0\mathrm{D}}$. For the diffusion equation, we use (multiple) time steps of a second order consistent Crank-Nicolson scheme with time step $dt_{1\mathrm{D}}$. We couple the stationary Bidomain Equation (3) uni-

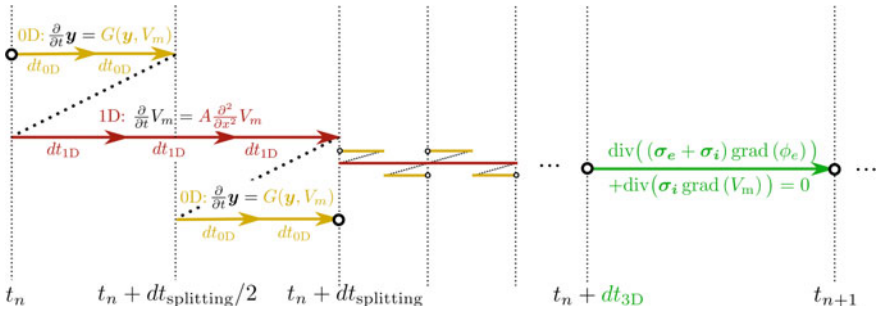


Fig. 1 Overall time stepping for the solution of the system of Eqs. (1)–(3). The diagram shows a sequence of Strang splitting steps for Eqs. (1) (yellow) and (2) (red) followed by a solution of Eq. (3) (green). Since Eq. (3) is stationary, it holds $t_n + dt_{3D} = t_{n+1}$. The complete sequence is called a global time step

directionally to the Monodomain system, Eqs. (1) and (2). After the sequence of Strang splitting steps the coupling variable V_m gets transferred from Ω_f^j to Ω_m. By solving the stationary Bidomain Equation (3), a new approximation for the extracellular potential ϕ_e is found, which represents the current EMG signal.

3 Implementation and Optimizations

We solve the model in our open-source software framework *opendihu*.[1] It allows to compose a nested structure of solvers and numerical schemes, and thus serves as a playground tool to identify optimal schemes for the problem at hand. It consists of a core C++ library that builds on various data management and solver packages, such as MPI, ADIOS2,[2] PETSc [2], MUMPS [1] and Hypre [5].

Formulated models, e.g., for subcellular kinetics, can be integrated using the community standard CellML [8], an XML based description. A simulation program for a specific model comprises compile-time and runtime code. A short C++ main file defines the graph of solvers by means of C++ templates at compile time, whereas at runtime, a Python script gets interpreted. It defines parameters and callback functions to modify them during the simulation. More details on the software structure are available in a previous publication [14].

The solution of the governing equations is suited for parallel computation, as has been discussed more detailed in [14]. The Monodomain Equations on all the muscle fibers are independent of each other. Furthermore, all instances of the subcellular Eq. (1) are independent of each other and can be computed concurrently. Only the computation on the 3D domain, Eqn. (3), involves information in the whole domain

[1] https://github.com/maierbn/opendihu/.

[2] https://github.com/ornladios/ADIOS2.

Ω_m. We partition the computational domain into disjoint subdomains for the available processes. As we use a structured 3D mesh, this can be done easily by defining appropriate index sets in x, y and z direction. The 1D muscle fiber meshes for Ω_f^j are partitioned in the same way as the 3D mesh for Ω_m, such that every process owns a distinct part of the overall spatial domain.

In this section, we focus on the compute-node level performance of the Monodomain system, Eqs. (1) and (2), as well as efficient in situ visualization.

3.1 CellML

CellML is an XML based description, established in the bioengineering community for subcellular models. Any CellML model abstracts the right hand side of a differential-algebraic equation by formulating the following function:

$$\begin{pmatrix} \texttt{rates} \\ \texttt{algebraics} \end{pmatrix} = \texttt{cellml}\,(\texttt{states}, \texttt{constants})\,. \tag{4}$$

The input consists of the vector of `constants` as well as the vector of `states` for which the corresponding vector of `rates` are computed. The vector of `algebraics` contains additional output values. Each scalar has a unique name in the CellML description, by which it is also identified by *opendihu*.

Various tools exist to manipulate and solve CellML models. A programming interface (API) as well as the CellML website provide means to use CellML models in various programming languages such as C, MATLAB and Python. The open-source tool OpenCOR [9] can be used to edit, simulate and analyze such models in a user-friendly graphical user interface. It focuses on single-cell models and, thus, is not able to simulate the Monodomain Equation. In the domain of cardiac electrophysiology simulations, Chaste [7] is a popular software framework. It provides a utility to convert CellML models for the use in the framework. A tool with similar purpose is Myokit [6]. Another software environment is OpenCMISS [3] which also can be used to simulate the Monodomain Equation.

Our solution differs from existing tools in that it is targeted at skeletal muscle simulations at high resolution on supercomputers. Through the tight integration in *opendihu*, specific optimizations are possible such as explicit vectorization. In our system comprising Eqs. (1) and (2), the `states` vector contains the transmembrane potential V_m as well as the subcellular state $\boldsymbol{y}$. All parameters such as the stimulation current I_stim are part of the `constants` vector. The `algebraics` part is empty for the present model of EMG simulation. It contains values in other scenarios, e.g., when muscle contraction is considered. If required, our tool is also capable of handling `algebraics`. Figure 2 summarizes the steps starting from a given CellML file to solving Eqs. (1) and (2) carried out by our software *opendihu*.

3.2 General Optimizations of the Code Generator

As shown in Fig. 2, a code generator produces optimized C code from a CellML problem description. The naive way to solve all the instances of a CellML problem on a process required for Eq. (2) leads to storing the state vectors in an Array-of-Struct (AoS) memory layout, where all state variables for a 0D unit or a 1D mesh point are close in memory. This approach is used, e.g., in OpenCMISS [3].

Using a Struct-of-Array (SoA) memory layout instead, where all instances of a state variable are contiguously stored, and additionally modifying the iteration loops such that the outer loop is over instances and the inner loop over the states allows cache-efficient memory access and to make use of the single-instruction multiple-data (SIMD) paradigm. We provide two specific types of code generation, "`comp`" and "`vc`". Both make use of the SIMD paradigm. The `comp`-code generator relies on the compilers capability of auto-vectorization by employing the available instruction set, e.g., leading to simultaneous computations of 4 double values with AVX2 or 8 double values with AVX-512. The Hazel Hen system uses the AVX2 instruction set with 4 double values.

Experiments show that relying on the compilers auto-vectorization does not lead to optimal results. I.e., for a system with AVX2, our measurements show that the runtime decreases approximately by a factor of two. Therefore, we additionally implement explicit vectorization within the `vc`-code generator. We use the C++ library *Vc*[3] which abstracts SIMD instructions and compiles code that uses the respective instruction sets. The used memory layout for this implementation is Array-of-Vectorized-Struct (AoVS) and the outer loop iterates over the CellML instances in groups according

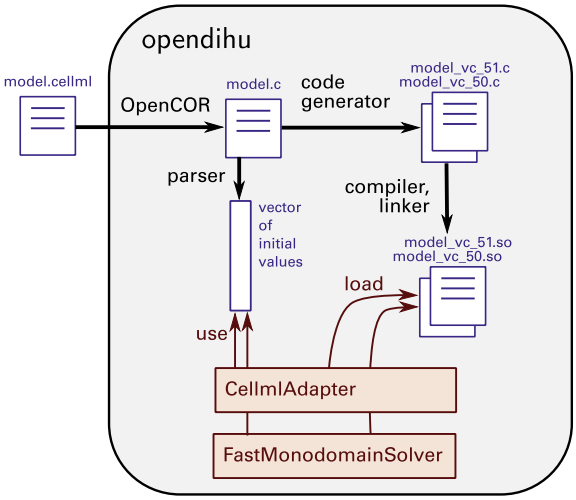


Fig. 2 Schematic representation of information flow for the CellML subsystem of *opendihu*. First, the command line interface of OpenCOR is used to convert the model to a C code file. The initial values for all states get parsed from this file and will be used later to initialize the state vector. Additionally, all compute instructions are parsed into an internal syntax tree. Then, the code generator produces optimized C code that can solve as many independent instances of the CellML model as are assigned to each MPI rank by the global domain decomposition

[3] https://vcdevel.github.io/Vc-1.4.1.

to the vector register length. Where the computation in the CellML model branches, i.e., for piecewise-defined functions, all cases have to be computed and the inactive values have to be masked out.

Profiling the code for different CellML models shows that about half the time is spent in evaluating the exponential function. Therefore we approximate

$$\exp(x) \approx \exp^*(x) = \left(1 + \frac{x}{n}\right)^n, \qquad n = 1024 = 2^{10}.$$

This computation can be performed using only 1 addition and 11 multiplications. In the considered CellML models, the values for x are bounded by $|x| < x_{\max} = 12$, for which the relative error is $|(\exp^* - \exp)(x_{\max})/\exp(x_{\max})| < 0.07$. Furthermore, we found that potentiation only occurs with exponents that are whole numbers. We replace the generic `pow` function by a more efficient, recursive implementation.

3.3 Specific Optimizations for the Monodomain Equation

Further specific performance improvements in *opendihu* are employed within the `FastMonodomainSolver` class. As shown in Fig. 2, the class also uses the code generator and replaces the `CellmlAdapter`. The implementation exploits the particular structure of the Monodomain Equation.

When using the general `CellmlAdapter`, the solution of the 1D diffusion problem in Eq. (2) is done using a parallel conjugate gradient (CG) solver of PETSc, which requires costly communication between processes. The `FastMonodomainSolver` exploits the fact that, for the whole muscle, we solve many 1D problems (one for each muscle fiber), each of them small enough to be solved sequentially with the simple linear complexity Thomas' algorithm. Since the 3D problem requires a three-dimensional domain decomposition as described in [14], all data required to solve the 0D-1D system for a given fiber have to be first communicated to one particular MPI rank (within a node). This rank then performs the 0D/1D computations serially for a time span of dt_{3D}. Afterwards, all data are communicated back to the original location, as needed for the subsequent computation of the 3D model. The selection of the process to do the respective computation for a given fiber follows a round-robin fashion such that all processes receive the same amount of workload.

The code for this computation is again generated by the code generator using the *Vc* library allowing arbitrary subcellular CellML models. Thus, the whole algorithm uses vector instructions and avoids unnecessary data copies. On the software side, this 'shortcut' algorithm partially specializes the same C++ class templates as the baseline version and therefore integrates well with the encompassing coupling scheme to the stationary Bidomain Equation. Also the same Python settings file can be used for both variants. This means that the change between the two schemes appears user-friendly in one line of code.

3.4 In Situ Visualization

As the large volume of simulation output in our envisaged research questions poses a challenge for a posteriori analysis by domain experts (see Sect. 1), we implemented and tested an in situ visualization setup. Besides reducing the data output, our setup has the capability to produce the image data on- or off-site as well as connecting the visualization to the VISUS Powerwall for interactive exploration of the high resolution visualization while the simulation is still running. We focus on the on-site visualization.

This means that one visualization instance is spawned on the same nodes the simulation is running on. The visualization instance collects all data produced on this node and renders a part of the final image. All partial images are then depth-correctly composited using IceT [16]. The nonexistence of GPUs on Hazel Hen made it necessary to rely on the CPUs for rendering. Thus, we implemented a new pipeline into the visualization framework MegaMol [11] that uses the ray tracing engine OSPRay [20] for rendering [17]. The original OpenGL-based rendering in MegaMol is performed in the corresponding renderer modules and implicitly composited via the default framebuffer. The new CPU rendering path processes data in the geometry modules and gathers all data in a single rendering module that owns the complete scene graph and therefore is able trigger a single render, global pass. Using ray tracing or optionally path tracing, global lighting and shading are available. These novel geometry modules can also be daisy-chained to easily compose complex scenes using a normal MegaMol project graph. However, in a distributed rendering scenario each instance of the visualization still generates a partial image from local data, which makes a composition step necessary to obtain the final image.

In general, we prefer the loosely-coupled in situ scheme [13], because this scheme increases the robustness of the joint simulation and visualization. Therefore, they do not run in the same MPI world and just communicate via small notification messages. However, this kind of scheme could not be realized on Hazel Hen, because executing two different programs on the same nodes is deactivated. As a fallback solution we run the setup in a single MPI world, where the simulation starts a visualization instance on each node (see Fig. 3). During initialization *opendihu* requests a connection with the local MegaMol instance via ZeroMQ[4] and checks if MegaMol has already initialized the module graph and is therefore ready for rendering. Depending on the communication pattern, each instance of *opendihu* that runs on the same node has to register at the MegaMol instance.

The connection uses MegaMol's built-in LUA interface [10] for notification and acknowledgments. After that, *opendihu* performs several simulation steps and writes data to the shared memory of the node. Upon finishing this, MegaMol gets notified with the explicit virtual 'file name'. MegaMol then maps the shared memory and processes the data for rendering. Note that a time step is synchronized in MegaMol, which means that MegaMol uses the information how many *opendihu* instances

[4] https://zeromq.org/.

run on a node to synchronize changing the displayed data to the latest complete simulation step.

The file format used for all transactions is based on bp-files as written by ADIOS2.[5] This offers a lot of flexibility to the setup and also enables parallel I/O. For example, the simulation can either handle every *opendihu* instance on a node independently or write a single file and limit the communication with MegaMol to a single instance per node. This reduces communication overhead, avoids potential connection issues and reduces the module graph overhead of MegaMol.

Connecting from anywhere to the head node of the visualization with another MegaMol instance starts the image transfer to the remotely connected instance. The on-site and off-site MegaMol instances also communicate via the LUA interface, enabling the user to interact with the visualization that is running on-site and transporting only images off-site.

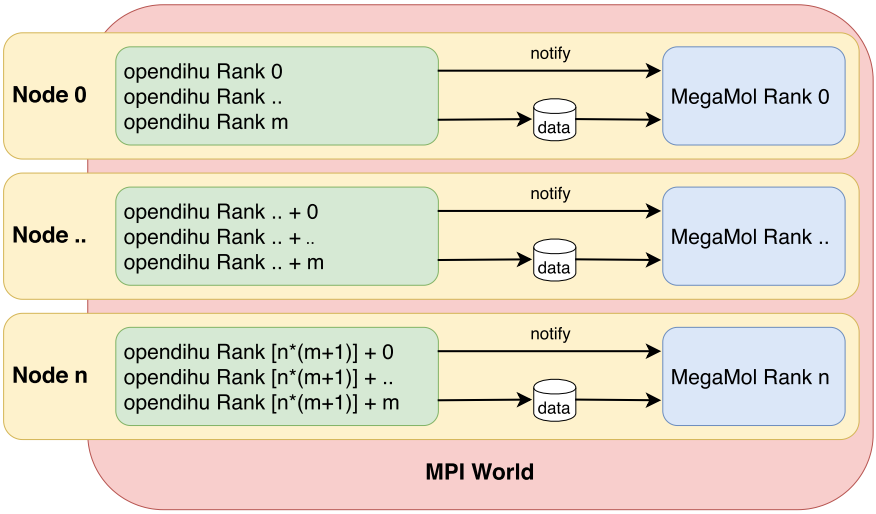


Fig. 3 Schematic of the in situ coupling. On each node a visualization instance is started and all *opendihu* processes communicate availability of their data once that is written into shared memory. The communication is a short message about new data from one or multiple simulation processes. To avoid rendering mixed images of old and new data, the visualization puts the new data on hold until all processes have signaled the availability of new data. Additionally, a connected client can send visualization parameters such as camera manipulation commands to the on-site visualization application, which is used for an interactive exploration of the data

[5] https://github.com/ornladios/ADIOS2.

4 Results and Discussion

In our evaluation, we use a realistic geometry of a Biceps Brachii muscle, which was extracted from the Visible Man dataset [19]. Figure 4 shows a simulation result with colored electric transmembrane potential V_m on multiple muscle fibers. Different improvements are analyzed in detail in the following.

The performance measurements are carried out on the supercomputer Hazel Hen at HLRS in Stuttgart. Per compute node, it contains two Intel Haswell E5-2680v3 CPUs at 2.5 GHz with 24 cores.

4.1 Choice of Compiler

At first, we investigate the auto-vectorization performance of the GNU, Intel and Cray compilers, and vary optimization levels both for the program and in the `CellmlAdapter`. We compute the system of Eqs. (1) and (2) with the subcellular model of Shorten et al. [18] for one muscle fiber with $n_p = 2400$ nodes, time step widths $dt_{0D} = 10^{-3}$ms, $dt_{1D} = dt_{\text{splitting}} = 3 \cdot 10^{-3}$ms and end time $t = 20$ ms. 24 processes are used on one compute node of Hazel Hen. The total runtimes for the 0D and 1D parts are depicted in Fig. 5.

It can be seen, that for all compilers computation times for the 0D model decrease if the optimization level is increased from `O1` to `O2` or `O3` and further to `Ofast`. The optimization levels `O2` and `O3` behave similarly for the given program. This shows that vectorization, which is already enabled by the `O2` flag, contributes most to the decrease in runtimes. For the same reason, the computation of the 1D model does not profit from a higher optimization level of the compiler. When using `Ofast`,

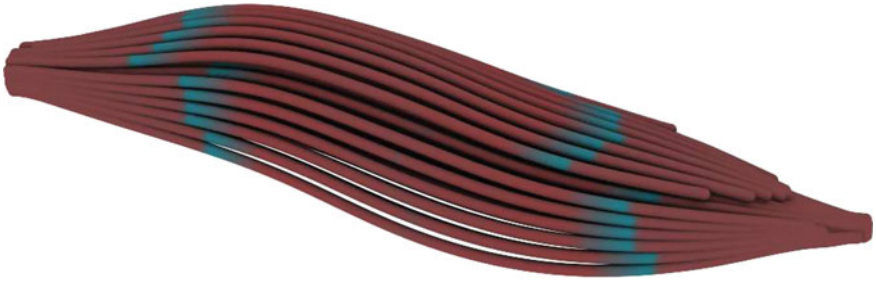


Fig. 4 Simulation result for multiple muscle fibers. The color represents the value of the electric transmembrane potential V_m

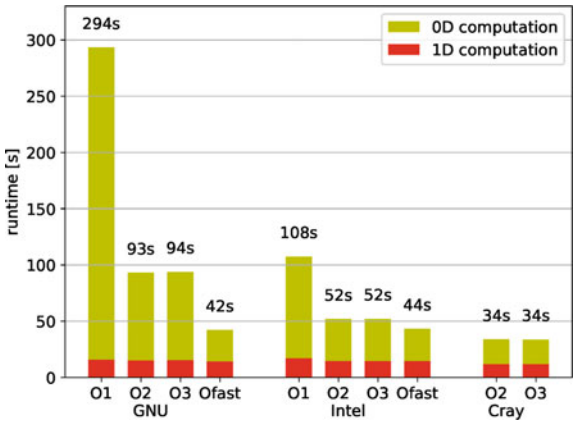


Fig. 5 Runtime of the 0D and 1D models with different compilers and optimization levels. The data is averaged over 120 runs

compliance regarding the IEEE and ISO rules for floating-point arithmetic and math functions gets disregarded, which means for example that infinite math is not supported. This is not an issue for our simulation code. As can be seen in the measurement of the GNU and Intel compilers, this optimization level yields a further increase in performance with more than 2 times faster code for the GNU compiler. Comparing the compilers, it can be seen, that, for the same optimization level, the Cray compiler produces faster code than the Intel compiler, which in turn produces faster code than the GNU compiler. The best result is achieved with the Cray compiler and the `O3` flag. However, the compile time increases to over 2 h compared to approximately 10 min with the other two compilers.

4.2 Monodomain Solver Improvements

In a small scale preliminary study, we measure the performance of the code generator and its explicit vectorization. We solve the Monodomain system, Eqs. (1) and (2), using the `FastMonodomainSolver` with the `vc`-code generator on the same scenario as in Sect. 4.1. We use two processes on an Intel Skylake I5-6300U dual-core processor with 2.4 GHz base frequency, which has hardware counters for scalar and vectorized floating point operations. During computation, we measure a performance of 19.6 GFlops which corresponds to 25.5% of the theoretical peak performance of 76.8 GFlops for this processor. This is an indication of good performance and results from the explicit SIMD instructions employed by the code generator.

The next study compares weak scaling runtimes for the Monodomain system solved with the `CellmlAdapter` and the `FastMonodomainSolver` with the subcellular model of Hodgkin-Huxley [12]. The results are depicted in Fig. 6. The number of processes increases from 2 to 6728, and the number of muscle fibers from 49 to 182,329. The spatial resolution within the fibers stays constant, with a

number of $n_{1D} = 1480$ elements per muscle fiber. We solve the complete model including the EMG, which requires communication of the respective quantities between the 3D and 0D/1D domains. For the sake of this study, only the runtimes of the 0D-1D solution as well as the additional time for communication in the `FastMonodomainSolver` are shown. The baseline solvers (dashed lines in Fig. 6) use the `CellmlAdapter` with the `vc`-code generator and a parallel CG solver of PETSc for the 1D diffusion along fibers. The improved method (solid lines) uses the `FastMonodomainSolver`.

We use time step widths of $dt_{0D} = 10^{-3}$ ms, $dt_{1D} = 2 \cdot 10^{-3}$ ms, $dt_{\text{splitting}} = 2 \cdot 10^{-3}$ ms and $dt_{3D} = 10^{-1}$ ms and simulate one and two time steps of the 3D and 0D-1D problems, respectively. To obtain the correct relation of these runtimes, we consider a simulation end time of 10^{-1} ms and scale the measured durations according to the number of time steps in this simulation time span.

The results show a sharp increase in runtime for the 1D baseline solver which is due to increased communication. The computation of the 0D subcellular model which involves no communication is constant for both schemes. Compared to the baseline, the improved algorithm shows a runtime speedup of approximately 8 for the 0D solver and between 2 and 19 for the 1D solver.

Additionally, the 1D baseline solver shows large variations between minimum and maximum solution times on different processors. The wall time of the simulation would correspond to the maximum solution times which contributes to an even larger speedup for our new solver. For the improved solver, these variations are effectively eliminated. The data transfer time is also part of the runtime for the new solver, but it is negligible as it occurs only twice in the large time interval of dt_{3D}.

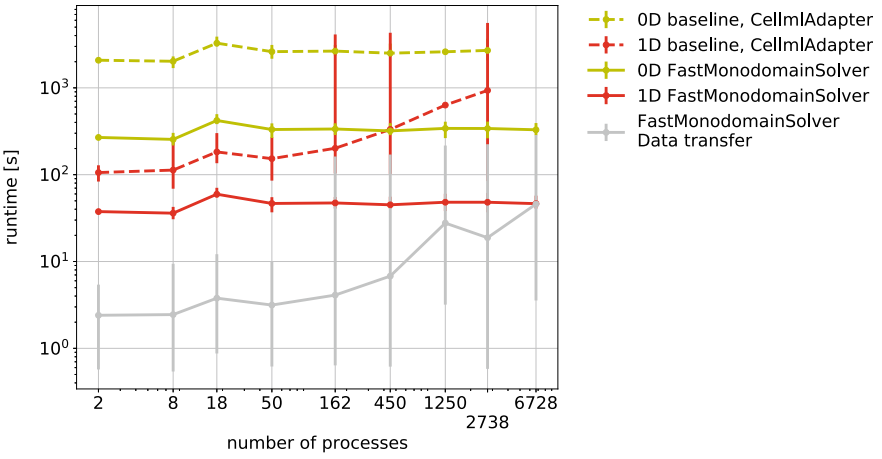


Fig. 6 Comparison of runtimes for solving the Monodomain system, Eqs. (1) and (2), using the `CellmlAdapter` (dashed lines) and the `FastMonodomainSolver` (solid lines). Depicted is the runtime in a weak scaling setup as well as indications for minimum and maximum values over all processes for each data point

4.3 Bidomain Solver Improvements

The weak scaling of the CG solver for the 3D stationary Bidomain Equation (3) has also been improved by updating the PETSc version used by *opendihu*. We mainly profit from an improved communication implementation in the newer PETSc version which affects the CG solver performance. Figure 7 shows the runtime of the CG solver on Hazel Hen for a fixed number of solver steps. The number of unknowns per process is relatively small, so that the runtime is dominated by communication time. With the new PETSc version, the increase in runtime with larger number of processes that initially followed approximately a square root function is replaced by the expected logarithmic growth.

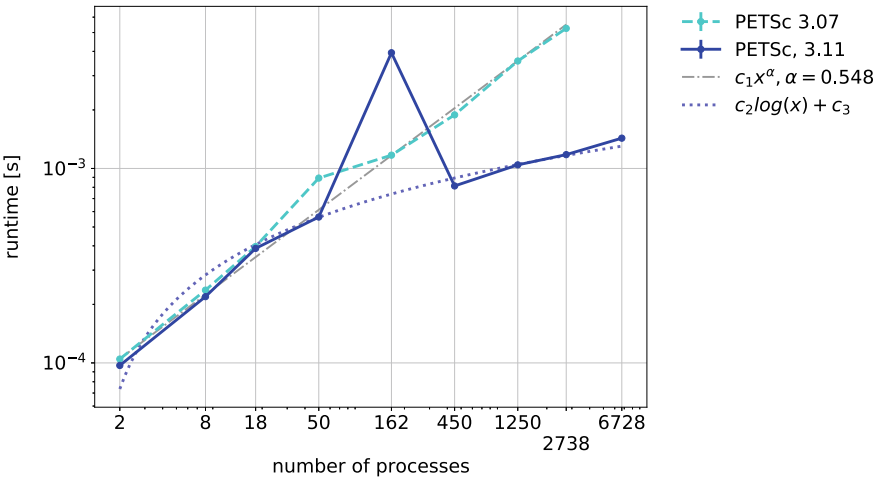


Fig. 7 Improvements in the CG solver due to newer PETSc version. Runtimes are shown for 3 CG iterations on Hazel Hen with approximately 248 unknowns per process. Each process is assigned to an unique core where all cores in a node are utilized. The peak at 162 processes is a reproducible outlier which we currently cannot attribute to a specific reason. To illustrate the different scaling behavior the dashed reference lines show fitted algebraic and logarithmic functions

4.4 Weak Scaling of the Overall Model

Taking into account all improvements, in Fig. 8 we present weak scaling results with the same setting as in the weak scaling study in Sect. 4.2. It shows the relation of runtimes for the 0D, 1D and 3D models as well as initialization time for a realistic simulation with end time 1 s. For the CG solver for the 3D problem, we prescribe a fixed number of 10^4 iterations. The 3D time step width of $dt_{3D} = 10^{-1}$ ms corresponds to a sampling frequency of a simulated EMG measurement device of 10 kHz.

It can be seen that the 0D solver consumes most of the runtime, despite extensive use of SIMD instructions, whereas the 3D solver has the lowest runtimes. The duration for initialization which mainly consists of reading the geometry data from input files increases with higher problem sizes, but remains significantly lower than the 0D and 1D solvers.

For the 3D solver, large spans between minimum and maximum runtimes can be seen, which are due to a very small number of processes that receive a remainder subdomain that is smaller than the average. In total, the 0D, 1D and 3D components of the simulation exhibit good weak scaling properties.

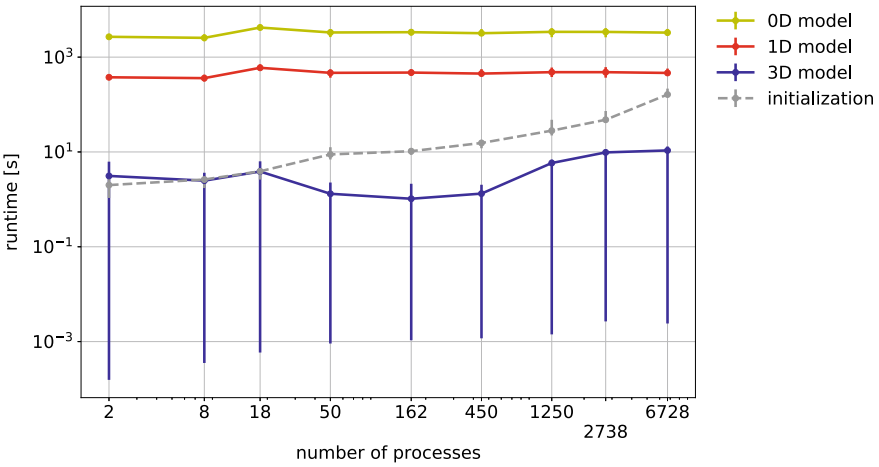


Fig. 8 Weak scaling results, runtimes of components of the simulation scaled to match a simulation time of 1 s. The mean runtime of the 0D,1D and 3D solvers as well as the duration for initial loading of the fiber geometry is shown, with bars indicating minimum and maximum over all processes

Table 1 Measurement of the I/O performance using three different file writers available in *opendihu*

Format	File size [Mb]	Output duration [s]
Exfile[a]	8800	311.06
VTK[b]	65	1.3
bp (ADIOS2)	33	0.68

[a] http://opencmiss.org/
[b] https://vtk.org/

4.5 In Situ Visualization

As a first step, we improved the I/O of the simulation to be compatible with the visualization framework and improve the parallel I/O performance. We simulate 1,369 fiber meshes and use 13,200 nodes in the 3D EMG mesh using 64 processes on an Intel Xeon E7-8880 v3 processor with 72 physical cores and SSD. The 3D mesh is written in parallel and at 40 time steps using the output file formats *Exfile*, *VTK* and *bp*. The *Exfile* format is a text-based format used by the OpenCMISS community. VTK is a base64-encoded, convenient format that can be visualized, e.g., by the program ParaView. The *bp* format of ADIOS2 is optimized for parallel output.

The total durations of writing the output and the resulting file sizes are summarized in Table 1. It can be seen that ADIOS2 clearly outperforms the other two formats both in file size and runtime. In the in situ scenario, where data is only communicated via shared memory in our loosely-coupled setting (see Sect. 3.4), the output time is expected to be even shorter.

Previous setups show that MegaMol runs stable on 128 nodes, more nodes have not been tested yet. However, on Hazel Hen only single-node proof-of-concept runs have been conducted. To test the setup for the scale of a full-sized simulation, we have to prepare further tests. While waiting for availability of the new supercomputer Hawk,[6] we recreated a small-scale setup with currently available hardware and adopted several new features from current MegaMol development, for example new interaction and abstraction patterns for camera control, a completely overhauled remote visualization component and a significantly extended ADIOS2 integration. Furthermore, the internal OSPRay data structures and internal communication patterns have been significantly improved. These features have been developed in parallel, but have not been tested at scale on an HPC system yet. In Fig. 9 a basic visualization is shown, while running in situ with the simulation. The small-scale setup consists of two Intel Xeon Phi 7210 that run the simulation and the visualization on-site. The memory that the simulation maps for rendering depends completely on the system size, the parallelization of the simulation and the number of buffered time steps. The visualization framework's memory overhead varies slightly, depending on the framebuffer size, the number of ranks because of the image composition and the communication pattern. In this particular case, we measured a memory overhead of 660 Mb for the on-site

[6] https://www.hlrs.de/systems/hpe-apollo-hawk/.

Fig. 9 In situ visualization of muscle fibers in the proof-of-concept in situ test case. In this case, we use a built-in *opendihu* simulation scenario with multiple muscle fibers and the mapped electric transmembrane potential, V_m. The simulation is connected to the visualization framework MegaMol

running visualization using a full HD resolution and a single communication channel per node. We consider this to be reasonable to do an on-site visualization, since we also expect no significant deviations at a larger scale. However, for optimization we need to conduct further tests on the new system.

5 Conclusion

We have presented scalability results for our new implementation of a neuromuscular system simulation. In particular, we could improve runtimes by up to a factor of 25 compared to previous results by optimal choices of compilers, optimized code generation exploiting AVX vectorization as well as the particular structure of the system of equations we are using. We enhanced the simulation by a communication-minimizing in situ visualization with MegaMol. Some results are preliminary due to the substantially reduced availability of Hazel Hen and Hawk during the transition to the new system Hawk throughout the last six months. Since the HLRS systems have become available again shortly before the submission deadline, we will perform full runs and include performance results and discussion for the camera-ready deadline. In particular, we will verify stability and performance of different communication patterns in the in situ visualization approach at scale on Hawk and make measurements of the

performance impact on the simulation and the additional memory consumption of the on-site visualization.

Acknowledgements We thank the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) for supporting this work by funding—EXC2075—390740016 under Germany's Excellence Strategy. We acknowledge the support by the Stuttgart Center for Simulation Science (SimTech). Further funded by the International Research Training Group "Soft Tissue Robotics" (GRK 2198/1).

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. P. Amestoy, A. Buttari, J.Y. L'Excellent, T. Mary, On the complexity of the block low-rank multifrontal factorization. SIAM J. Sci. Comput. **39**(4), 1710–1740 (2017). (16M1077192)
2. S. Balay, W.D. Gropp, L.C. McInnes, B.F. Smith, Efficient management of parallelism in object oriented numerical software libraries, in *Modern Software Tools in Scientific Computing*, ed. by E. Arge, A.M. Bruaset, H.P. Langtangen (Birkhäuser Press, 1997), pp. 163–202
3. C. Bradley, A. Bowery, R. Britten, V. Budelmann, O. Camara, R. Christie, A. Cookson, A.F. Frangi, T.B. Gamage, T. Heidlauf, S. Krittian, D. Ladd, C. Little, K. Mithraratne, M. Nash, D. Nickerson, P. Nielsen, O. Nordbo, S. Omholt, A. Pashaei, D. Paterson, V. Rajagopal, A. Reeve, O. Röhrle, S. Safaei, R. Sebastián, M. Steghöfer, T. Wu, T. Yu, H. Zhang, P. Hunter, OpenCMISS: a multi-physics & multi-scale computational infrastructure for the VPH/Physiome project. Prog. Biophys. Mol. Biol. **107**(1), 32–47 (2011). https://doi.org/10.1016/j.pbiomolbio.2011.06.015
4. C.P. Bradley, N. Emamy, T. Ertl, D. Göddeke, A. Hessenthaler, T. Klotz, A. Krämer, M. Krone, B. Maier, M. Mehl, R. Tobias, O. Röhrle, Enabling detailed, biophysics-based skeletal muscle models on HPC systems. Front. Physiol. **9**(816) (2018). https://doi.org/10.3389/fphys.2018.00816
5. J. Brannick, F. Cao, K. Kahl, R.D. Falgout, X. Hu, Optimal interpolation and compatible relaxation in classical algebraic multigrid. SIAM J. Sci. Comput. **40**(3), A1473–A1493 (2018). https://doi.org/10.1137/17M1123456
6. M. Clerx, P. Collins, E. de Lange, P.G. Volders, Myokit: a simple interface to cardiac cellular electrophysiology. Progr. Biophys. Mol. Biol. **120**(1), 100–114 (2016). https://doi.org/10.1016/j.pbiomolbio.2015.12.008
7. J. Cooper, R.J. Spiteri, G.R. Mirams, Cellular cardiac electrophysiology modeling with Chaste and CellML. Front. Physiol. **5**, 511 (2015). https://doi.org/10.3389/fphys.2014.00511
8. A.A. Cuellar, C.M. Lloyd, P.F. Nielsen, D.P. Bullivant, D.P. Nickerson, P.J. Hunter, An overview of CellML 1.1, a biological model description language. SIMULATION **79**(12), 740–747 (2003). https://doi.org/10.1177/0037549703040939
9. A. Garny, P.J. Hunter, OpenCOR: a modular and interoperable approach to computational biology. Front. Physiol. **6**, 26 (2015). https://doi.org/10.3389/fphys.2015.00026
10. P. Gralka, M. Becher, M. Braun, F. Frieß, C. Müller, T. Rau, K. Schatz, C. Schulz, M. Krone, G. Reina, T. Ertl, MegaMol—a comprehensive prototyping framework for visualizations. Eur. Phys. J. Spec. Top. **227**(14), 1817–1829 (2019). https://doi.org/10.1140/epjst/e2019-800167-5
11. S. Grottel, M. Krone, C. Müller, G. Reina, T. Ertl, MegaMol—prototyping framework for particle-based visualization. IEEE Trans. Visual Comput. Graphics **21**(2), 201–214 (2015). https://doi.org/10.1109/TVCG.2014.2350479

12. A.L. Hodgkin, A.F. Huxley, A quantitative description of membrane current and its application to conduction and excitation in nerve. J. Physiol. **117**(4), 500–544 (1952)
13. J. Kress, S. Klasky, N. Podhorszki, J. Choi, H. Childs, D. Pugmire, Loosely coupled in situ visualization: a perspective on why it's here to stay, in *Proceedings of the First Workshop on In Situ Infrastructures for Enabling Extreme-Scale Analysis and Visualization, ISAV2015* (Association for Computing Machinery, 2015), pp. 1–6. https://doi.org/10.1145/2828612.2828623
14. B. Maier, N. Emamy, A.S. Krämer, M. Mehl, Highly parallel multi-physics simulation of muscular activation and EMG, in *Coupled Problems 2019* (2019), pp. 610–621
15. M. Mordhorst, T. Heidlauf, O. Röhrle, Predicting electromyographic signals under realistic conditions using a multiscale chemo-electro-mechanical finite element model. Interface Focus **5**(2), 1–11 (2015). https://doi.org/10.1098/rsfs.2014.0076
16. K. Moreland, W. Kendall, T. Peterka, J. Huang, An image compositing solution at scale, in *Proceedings of 2011 International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2011* (ACM, 2011), pp. 25:1–25:10. https://doi.org/10.1145/2063384.2063417
17. T. Rau, M. Krone, G. Reina, T. Ertl, Challenges and opportunities using software-defined visualization in MegaMol, in *Workshop on Visual Analytics, Information Visualization and Scientific Visualization (WVIS) in the 30th Conference on Graphics, Patterns and Images (SIBGRAPI 2017)*, ed. by N. Ferreira, L.G. Nonato, F. Sadlo (2017)
18. P.R. Shorten, P. O'Callaghan, J.B. Davidson, T.K. Soboleva, A mathematical model of fatigue in skeletal muscle force contraction. J. Muscle Res. Cell Motil. **28**(6), 293–313 (2007)
19. V. Spitzer, M.J. Ackerman, A.L. Scherzinger, D. Whitlock, The visible human male: a technical report. J. Am. Med. Inform. Assoc. **3**(2), 118–130 (1996). https://doi.org/10.1136/jamia.1996.96236280
20. I. Wald, G. Johnson, J. Amstutz, C. Brownlee, A. Knoll, J. Jeffers, J. Günther, P. Navratil, OSPRay—a CPU ray tracing framework for scientific visualization. IEEE Trans. Vis. Comput. Graph. **23**(1), 931–940 (2017). https://doi.org/10.1109/TVCG.2016.2599041

Reactive Flows

Dietmar Kröner

In this section we have two contributions:

"Implementation of an Efficient Synthetic Inflow Turbulence-Generator in the Open-Source Code OpenFOAM for 3D LES/DNS Applications"

by Flavio Cesar Cunha Galeazzo, Feichi Zhang, Thorsten Zirwes, Peter Habisreuther, Henning Bockhorn, Nikolaos Zarzalis, and Dimosthenis Trimis

and

"Implementation of Lagrangian Surface Tracking for High Performance Computing"

by Thorsten Zirwes, Feichi Zhang, Jordan A. Denev, Peter Habisreuther, Henning Bockhorn, and Dimosthenis Trimis

Both papers are related to software which is based on OpenFOAM. They have added some own software tools to the OpenFOAM code and could improve the performance compared to the standard OpenFOAM tools.

In the first paper a new Synthetic Turbulent Inflow (STI) boundary condition in the framework of OpenFOAM is presented. It turns out that this new inflow boundary condition can generate a realistic turbulent flow field at a given inlet plane. The method is based on convolution of digital random data series. The bulk flow rate, turbulence intensity, turbulent length and time scales can be specified. Compared to previous implementations, the new turbulence generator is computationally more efficient by using coarse virtual grids and can be used for arbitrarily shaped inlets. While the OpenFOAM's native turbulence generator, shows some anomalies during parallel runs, the new implementation gives consistent results even for large-scale parallel simulations. The new boundary condition has been applied to turbulent combustion with Large Eddy Simulation (LES) and Direct Numerical Simulation (DNS). The main advantage of this new STI is first , that the

D. Kröner
Abteilung für Angewandte Mathematik, Universität Freiburg
Hermann-Herder-Str-10, 79104 Freiburg, Germany
e-mail: dietmar.kroener@googlemail.com

computational efficiently is greatly increased by using coarse virtual grids for the generation of the inlet boundary velocity field, second, arbitrarily shaped inlets can be trearted in the simulation and third, that OpenFOAM's native turbulence generator shows inconsistencies in parallel simulations. The parallel scaling is super-linear up to about 8,000 computational cells per MPI rank on Hawk.

This paper is supported by the Helmholtz Association of German Research Centers (HGF).

In the second contribution about "Implementation of Lagrangian Surface Tracking for High Performance Computing" a Lagrangian tracking algorithm is applied to the direct numerical simulation of 3D turbulent flames. The algorithm performs the tracking of material points on iso-surfaces and is implemented in OpenFOAM. The use of an automatically generated, highly optimized code for the computation of chemical reaction rates implies, that the code performs up to 20 times faster than the best settings for the standard OpenFOAM solver without loss of accuracy. The tracking method works on unstructured meshes with arbitrary cell shapes. The code is written in a general way and not restricted to reacting flows but can be used to track any iso-surface. The parallel scaling of the combustion solver is nearly ideal with and without Lagrangian tracking.

This work was supported by the Helmholtz Association of German Research Centres (HGF) and the DFG ("Deutsche Forschungsgemeinschaft").

Implementation of an Efficient Synthetic Inflow Turbulence-Generator in the Open-Source Code OpenFOAM for 3D LES/DNS Applications

Flavio Cesar Cunha Galeazzo, Feichi Zhang, Thorsten Zirwes, Peter Habisreuther, Henning Bockhorn, Nikolaos Zarzalis, and Dimosthenis Trimis

Abstract A new inflow boundary condition (BC) has been implemented into the open-source CFD code OpenFOAM, which generates synthetic turbulent fluctuations at the inlet boundary for 3D transient simulations. The method is based on convolution of digital random data series. The filter coefficients of the convolution process prescribe a two-point correlation function that possesses the basic properties of real turbulent flow. In this way, spatially and temporally correlated flow fields with specified bulk flow rate, turbulence intensity, turbulent length and time scales can be generated. Compared to previous implementations, the new turbulence generator is computationally more efficient by using coarse virtual grids and can be used for arbitrarily shaped inlets. Compared to OpenFOAM's native turbulence generator, which shows some anomalies during parallel runs, the new implementation gives consistent results even for large-scale parallel simulations. The inlet BC has been applied to two turbulent combustion cases with Large Eddy Simulation (LES) and Direct Numerical Simulation (DNS), using up to 8192 CPU cores on Hazel Hen at HLRS. The results reveal the significance of the inflow turbulence for reproducing the correct flame structure. A performance analysis of intra and inter-node performance on the Vulcan and Hawk clusters shows that the OpenFOAM solver is memory bound. Therefore, higher performance is reached when only half of the AMD CPU cores per node are utilized on Hawk because the L3 cache is shared by a core complex (CCX) and each core has a relatively low bandwidth. The simulation scales super-linearly on Hawk and reaches ideal speedup down to 8 000 computational cells per MPI rank, which is consistent with scaling results on the previous system Hazel Hen. The implementation of the BC is described in full detail.

F. C. C. Galeazzo (✉)
High Performance Computing Center Stuttgart (HLRS), Nobelstr. 19, 70569 Stuttgart, Germany
e-mail: flavio.galeazzo@hlrs.de

F. Zhang (✉) · T. Zirwes · P. Habisreuther · H. Bockhorn · N. Zarzalis · D. Trimis
Engler-Bunte-Institute/Combustion Technology, Karlsruhe Institute of Technology,
Engler-Bunte-Ring.7, 76131 Karlsruhe, Germany
e-mail: feichi.zhang@kit.edu

T. Zirwes
Steinbuch Centre for Computing, Karlsruhe Institute of Technology,
Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_14

Keywords Turbulent inflow generator · Turbulent combustion · DNS · OpenFOAM

1 Introduction

Properly chosen Boundary Conditions (BC) are of great importance in the simulation of flow systems. In the particular case of turbulent flows using the Direct Numerical Simulation (DNS) or Large Eddy Simulation (LES) approaches, a time-resolved velocity field is required for the inlet BC [1]. Because DNS and LES are transient simulation techniques by nature, in which all or a large part of the turbulent fluctuations are directly solved, DNS and LES require a transient, fluctuating velocity field to be prescribed at the inlet flow.

The simplest solution would be to use only a mean velocity, without fluctuations. Unfortunately, this method leads to unphysical flow behaviour with a much-delayed transition to turbulence [2]. An elegant way is to use periodic boundary conditions, where a plane is selected inside the computational domain and the resolved fluctuations from this plane are extracted and rescaled before being finally reintroduced as fluctuations to the mean velocity profiles at the inlet [3]. The turbulence generated by this method is very accurate; however, it is limited to simple geometries. Furthermore, the computational costs of this method are often higher than other alternatives [4]. Another simple method is to use a mean velocity profile with superimposed random fluctuations, where a random signal is added to the averaged velocity profiles to mimic fluctuations [5]. Although being very simple, in this method the energy of the fluctuations is equally distributed over the whole wavenumber range, which results in far too much energy in the high wavenumber, under-resolved scale and a lack of energy in the low wavenumber, resolved range. The fluctuations generated in this way are almost immediately dampened and the results are almost identical to using a mean velocity profile without fluctuations, i.e., laminar.

To avoid these problems, Klein et al. [2] proposed the use of digital filters and to rescale the random noise to generate fluctuations with a given spectrum. The technique has been successfully applied in many LES and DNS [6–11]. However, implemented in the way proposed by Klein et al. [2], the method suffers from some drawbacks, as very high computational cost at high resolutions, limitations of the inlet geometry due to the requirement of an equidistant, structured grid and spurious pressure fluctuations. These drawbacks are important for the applications we are interested in, i.e., LES and DNS using the compressible formulation of the governing equations of complex flows with a focus on turbulent mixing, combustion, acoustics and noise prediction, in which the geometries are often complex and the added computational cost often can reach prohibitive levels.

Some attempts in addressing these points can be found in the literature. Kempf et al. [12] propose modifications in the filtering operation and the generation of the random noise, reducing the computational cost and the memory requirements of the original version. Di Mare et al. [13] extended the method of Klein et al. [2] to

reproduce specified statistical data. The current work follows the same objectives, i.e., to reduce the computational cost and the memory requirements of the method, although using an alternative method. Also, the reduction of the unphysical pressure fluctuations generated by the original method has been investigated. The focus of the work is to introduce a new implementation of the inflow turbulence generator proposed by Klein et al. [2] in the open-source CFD code OpenFOAM, which is efficient and simple to use for LES and DNS simulations and gives consistent results for large-scale parallel simulations.

2 Mathematical Description

The Synthetic Inflow Turbulence (SIT) generator is based on the filtering procedure proposed by Klein et al. [2]. The filter uses a series of uncorrelated, random data r_m which has a mean value of zero and a standard deviation of one:

$$u_m = \sum_{n=-N}^{N} b_n r_{m+n}, \quad \text{with} \quad \sum_{m=1}^{M} r_m = 0 \quad \text{and} \quad \frac{1}{M}\sum_{m=1}^{M} r_m r_m = 1 \tag{1}$$

where u_m defines a convolution or a digital linear non-recursive filter of the series with the filter coefficient b_n. M and N are the length of the series and the number of support points used for the filter. Applying a two-point correlation function to u_m leads to:

$$R_{uu}(k) = \frac{\overline{u_m u_{m+k}}}{\overline{u_m u_m}} = \sum_{j=-N+k}^{N} b_j b_{j-k} / \sum_{j=-N}^{N} b_j^2 \tag{2}$$

with $\bar{\cdot}$ defining an ensemble averaged value. In order to calculate the filter coefficients b_n in Eq. (1), the correlation function for R_{uu} in Eq. (2) is set to have a presumed shape with

$$R_{uu}(r) = \exp(-\frac{\pi r^2}{4L^2}) \tag{3}$$

where r is the distance and L the length scale. If the distance r and the length scale L in Eq. (3) are replaced by the grid space Δx, i.e. $r = k\Delta x$ and $L = n\Delta x$, Eq. (2) and Eq. (3) lead to:

$$\frac{\overline{u_m u_{m+k}}}{\overline{u_m u_m}} = R_{uu}(k) = \exp\left(-\frac{\pi k^2}{4n^2}\right) \tag{4}$$

In this way, an explicit representation of the filter coefficients can be found by a multi-dimensional Newton method [2]:

$$b_k = \widetilde{b}_k / (\sum_{j=-N}^{N} \widetilde{b}_j^2)^{\frac{1}{2}} \quad \text{with} \quad \widetilde{b}_k = \exp(-\frac{\pi k^2}{4n^2}) \tag{5}$$

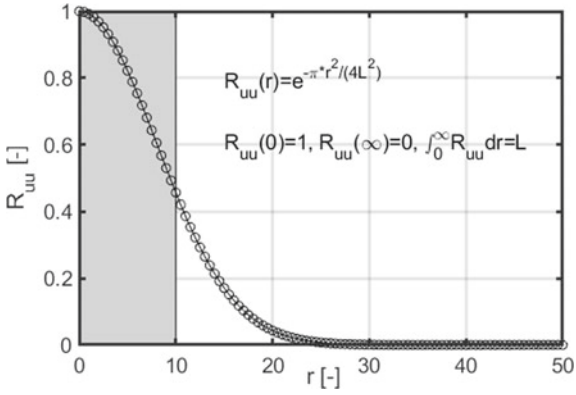


Fig. 1 Correlation function for determining the filter coefficients of the random data

This selected functional dependence by Eq. (3) fulfills the basic properties of a realistic two-point correlation function for homogeneous turbulence, i.e.:

$$R_{uu}(r=0)=1, \quad R_{uu}(r=\infty)=0, \quad \text{and} \quad \int_0^\infty R_{uu}(r)\,\mathrm{d}r = L \tag{6}$$

As an example, Fig. 1 plots the profile of $R_{uu}(r)$ against r by using $L = 10$, where the shaded area corresponds to the area under the curve. The predefined value with $L = 10$ is reproduced exactly by the integration $\int_0^\infty R(r)\,\mathrm{d}r$. A three-dimensional filter can be obtained by convolution of three one-dimensional filters to find the coefficients:

$$b_{i,j,k} = b_i b_j b_k \tag{7}$$

The filter operation over a three-dimensional random field R can then be calculated by

$$U(j,k) = \sum_{i'=-N_x}^{N_x} \sum_{j'=-N_y}^{N_y} \sum_{k'=-N_z}^{N_z} b(i',j',k') * R(i', j+j', k+k') \tag{8}$$

leading to the correlated data U with respect to the three spatial coordinates. The final velocity fluctuations for each spatial component are calculated as:

$$u_i = u_m + Tu \cdot u_{bulk} \cdot U_i \tag{9}$$

where u_m is the time-mean velocity field, which can assume arbitrary profiles such as flat-top or parabolic. u_{bulk} is the bulk flow velocity, which is evaluated from the time-mean velocity field. Tu is the turbulence level. The product of Tu and u_{bulk} provides the time-mean turbulence intensity (in m/s). U_i is the correlated data series produced by the introduced algorithm for the i-th spatial direction.

This technique for generating the inflow-turbulence synthetically has been implemented to the OpenFOAM flow solver [14] and issued for the current work. The reader is referred to the original work by Klein et al. [2] for more details regarding mathematical basics of the method.

3 Implementation

The algorithm proposed by Klein et al. [2] suffers from some drawbacks:

1. the computational cost can be very high at high resolutions;
2. the geometry of the inlet has to be of quadrilateral shape (because of the equidistant, structured grid);
3. spurious pressure fluctuations can affect the convergence negatively.

As pointed out in the introduction, some attempts of addressing these points can be found in the literature [12, 13]. Our approach, although very simple, has the advantage of addressing at the same time the computational cost and the utilization of the method in complex geometries. Instead of creating the fluctuations in a grid that exactly resembles shape and resolution of the numerical domain, a new grid is created which encompasses the inlet geometry, no matter how complex it is. The fluctuations are generated in this new grid, which can have an arbitrary resolution, and then interpolated onto the inlet grid. Figure 2 illustrates as an example the grid where the fluctuations are generated and an arbitrary corresponding inlet grid.

Using this method, the fluctuations can be generated with a coarser resolution, without compromising the quality of the artificial turbulence, which can reduce significantly the computational cost, as long as the coarse grid resembles the fluctuations by the Nyquist sampling theorem.

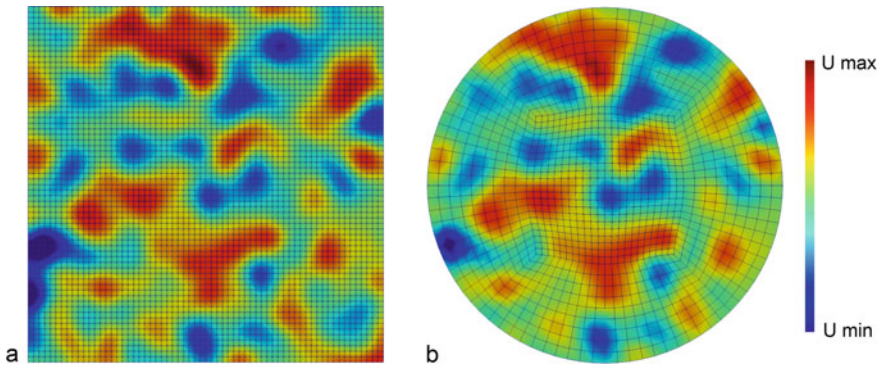


Fig. 2 Illustration of the interpolation method. The fluctuation velocity generated in the structured grid **a** is interpolated to the inlet boundary condition of the LES, regardless of its geometry **b**

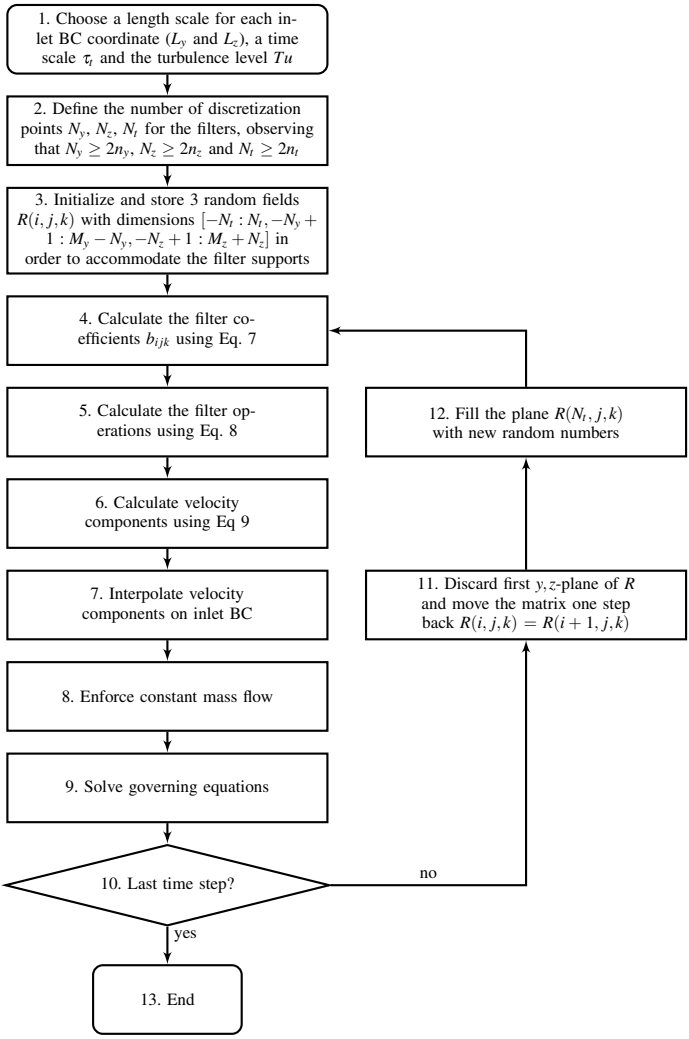


Fig. 3 Flow diagram of the algorithm

The spurious pressure fluctuations observed when applying the original algorithm are caused by small fluctuations of the mass flow induced by the method, which has been also observed by other authors [12]. These mass flow fluctuations are present even if the synthetic fluctuations have zero mean when the mean velocity profile is not constant. The solution is to apply a correction to the velocity field in every time step, to ensure constant mass flow. Some pressure fluctuation remains, as the synthetic turbulence field is not divergence-free. The amplitude of these pressure fluctuations is nevertheless small and they are usually readily dissipated.

Figure 3 shows the flow diagram, representing the implementation of the code in OpenFOAM. Observe that L_y, L_z and τ_t are the desired length and time scales in

physical space, D_y and D_z the size of the BC, U_{mean} the mean velocity perpendicular to the BC and Δt the time step. The filter frame has a resolution of $M_y \times M_z \times N_t$ elements. A restriction of the method is that the parameters N_y, N_z and N_t have to be at least twice their respective filter lengths, given by $n_y = M_y L_y / D_y, n_z = M_z L_z / D_z$ and $n_t = \tau_t / (U_{mean} \Delta t)$, approximated to the closest integer.

3.1 Validation

Figure 4 shows a comparison of the axial velocity component U_x and its autocorrelation R_{uu} using different filter frame resolutions and different generators. A simple incompressible LES has been set up, with a cubic domain with size $0.1 \times 0.1 \times 0.1$ m^3 and resolution of $64 \times 64 \times 64$ hexahedral elements. The inlet ($U_{mean} = 1$ m/s and $Tu = 10\,\%$) and the outlet conditions are along the x-direction, with periodic conditions otherwise. The subgrid-scale fluxes are modelled using the Smagorinsky model [1]. A monitor point recorded the axial velocity component U_x at the centroid of the domain for 10 s, where the first 2 s have been discarded from the analysis. The synthetic turbulence has been generated with filter frame resolutions $M_y = M_z = 64$ and 32, both with the same input for integral length scale $L = 0.009375$ m, marked by the grey area in Fig. 4. Using our Synthetic Inflow Turbulence generator (SIT), the agreement of the profiles is good, indicating that most of the quality of the turbulence is preserved when using coarser filter support for the turbulence generation. For comparison, results using the OpenFOAM native generator with the same parameters are also shown (turbulentDigitalFilterInlet, marked OpenFOAM in Fig. 4). The results are in good agreement with our proposed algorithm. It should be noted that our SIF model requires far less input information than the OpenFOAM implementation, making it easier to use.

One interesting result is that, when using the OpenFOAM generator in a parallel run, the pattern of the turbulent structures follows the partitioning of the domain

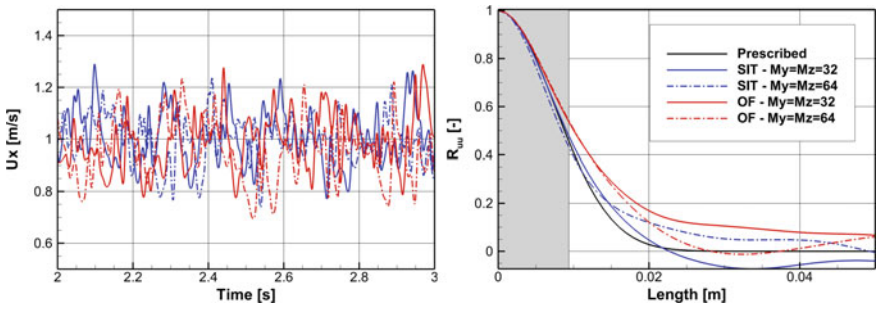


Fig. 4 Axial velocity component U_x and its autocorrelation R_{uu} using the new SIT and OpenFOAM's (OF) native synthetic turbulence generators.

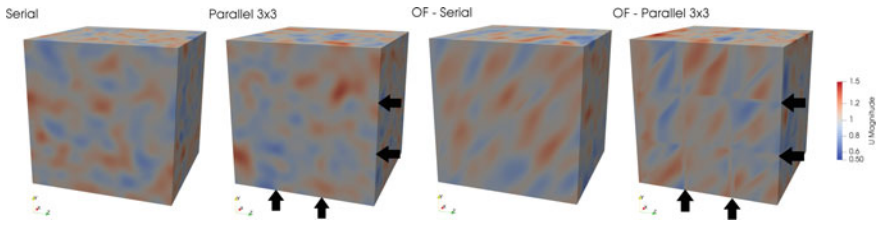


Fig. 5 Comparison of turbulent structures generated by serial and parallel computations using the new SIT and OpenFOAM's (OF) model.The arrows show the position of the parallel partitions

decomposition (process borders are indicated in Fig. 5 by the arrows), which may indicate a bug in the parallel implementation of the native OpenFOAM BC. Our algorithm implements its own interpolation routines and therefore the results are independent of the way the computational domain is decomposed.

3.2 Performance

The performance of the code has been assessed in the High Performance Computing (HPC) systems Vulcan and Hawk from the High Performance Computing Center Stuttgart (HLRS) [15]. Vulcan is a general-purpose system with nodes with different architectures. In this benchmark, nodes using CPUs Intel Xeon Gold 6138 at 2.0 GHz (Skylake, SKL) and 40 cores per node have been employed. Hawk is the supercomputer of HLRS and its nodes have AMD EPYC 7742 at 2.25 GHz (Rome) CPUs summing up to 128 cores per node. For this benchmark, the same simulation of Sect. 3.1 with an increased resolution of $128 \times 128 \times 128$ elements has been used, employing OpenFOAM 7 compiled with gcc 7.3 (Vulcan) and gcc 9.2 (Hawk), and OpenMPI as MPI package.

In this simple simulation, using a filter frame resolution of 32×32, the time penalty using the SIT BC is 11 % in comparison with using a periodic BC. The penalty is frequently lower in more complex simulations e.g. with combustion, as much more computational time is spent on computing chemical reaction rates and solving additional equations for the chemical species.

Figure 6a shows the intra-node performance relative to using all cores of the node. The Vulcan nodes exhibit the expected behaviour of Intel CPUs, with the highest performance obtained using all cores. On the other hand, the Hawk node shows the best results using 64 MPI ranks (half of the total available cores) by a good margin of over 40 % more performance. Because OpenFOAM applications are typically memory bound, the limiting factor is the memory bandwidth, which is lower per core compared to e.g. the previous system Hazel Hen. In addition, the L3 cache on the AMD CPU type is shared among four cores or one core complex (CCX), respectively.

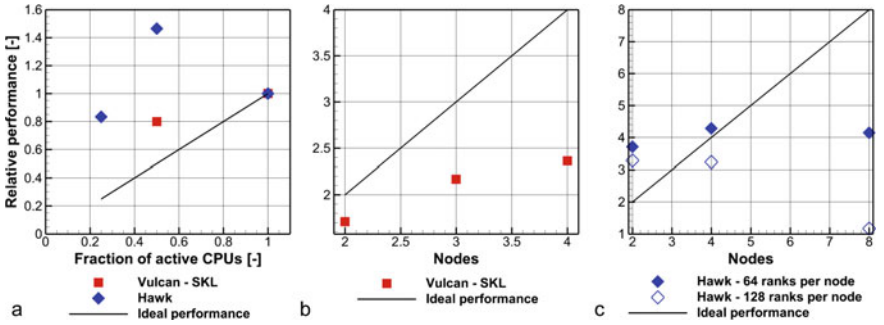


Fig. 6 Assessment of intra-node performance **a** and inter-node performance of the Vulcan **b** and Hawk **c** HPC systems

The inter-node performance of Vulcan relative to one single node can be seen in Fig. 6b, where the results show a poor scaling efficiency. The interesting results with Hawk are shown in Fig. 6c for two setups: using 128 and 64 ranks per node (all or one half the cores, respectively). Both configurations show super-linear speedup (relative performance above 100%) using 2 nodes, and using 64 ranks per node the speedup is super-linear even using 4 nodes. The efficiency is still approximately 50% using 8 nodes and 64 ranks per node, however, drops significantly using all core of the nodes.

As shown in a previous work [16], the simple flow solvers of OpenFOAM tend to scale reasonably well up to about 10 000 computational cells per MPI rank. Because this is a small test case, this limit is reached at two full nodes or four half nodes on Hawk, which correspond to about 8000 cells per MPI rank. There, the parallel efficiency is nearly ideal, confirming the results measured of the previous supercomputer Hazel Hen.

4 Applications

4.1 Numerical Setups

In order to demonstrate the capability of the method described in Sect. 2 and Sect. 3, it has been applied to simulate two turbulent combustion cases within the framework of Large Eddy Simulation (LES) and Direct Numerical Simulation (DNS).

- **LES of a premixed Bunsen-type flame** [17]**:** A natural gas flame operated at atmospheric pressure and a thermal load of $Q_{th} = 275.6$ kW is used, where the unburnt mixture has an equivalence ratio of $\Phi = 0.57$ (lean premixed) and is preheated to $T_0 = 400$ °C. The Reynolds number based on the nozzle diameter ($D = 0.15$ m) and bulk velocity ($v_{bulk} = 20.7$ m/s) is $Re \approx 48,000$. As illustrated

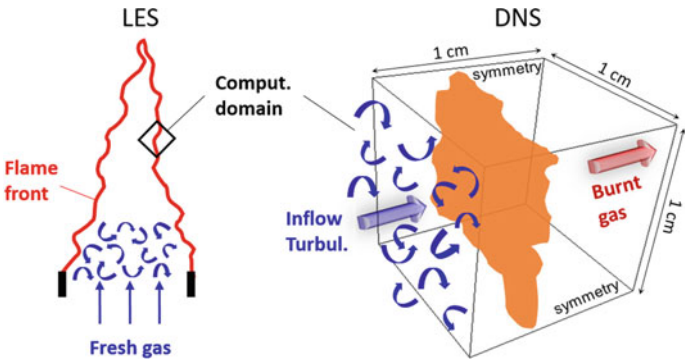


Fig. 7 Schematic drawing of the setups used for LES of a premixed Bunsen-type flame (left) and DNS of a synthetic freely-propagating flame (right).

in Fig. 7 on the left, fresh gas mixture enters the domain from the inlet at the bottom and is ignited, leading to a turbulent jet flame stabilized at the burner nozzle. The computational grid used for the LES case consists of approx. 6.3 million finite cells, which are systematically refined towards the shear layer of the jet and the flame surface, having a smallest grid resolution of $\Delta_{\min} = 0.5$ mm. The inlet plane is located $1D$ upstream of the burner exit.

- **DNS of a freely-propagating flame:** The second case for DNS is given by a 3D synthetic flame front which propagates freely in a cubic domain with the size of $1 \times 1 \times 1\,\mathrm{cm}^3$. As shown in Fig. 7 on the right, the setup can be considered as a small segment of a real flame front, which is sustained by a continuous flow of fresh gas. Along the streamwise direction, methane/air mixture enters the domain from the left at the inlet with $\Phi = 0.9$, $T = 300$ K and $p = 1$ bar. The product gas leaves the domain on the other side (outlet). The lateral faces are defined as symmetry planes to avoid a loss of mass. An equidistant computational grid with 64 million cells and a resolution of $\Delta_x = \Delta_y = \Delta_z = 25\ \mu$m is used for the DNS case, which can resolve the flame thickness ($\approx$0.5 mm) with approx. 20 grid points.

For both cases, the balance equations for total mass, momentum, energy are solved in their compressible formulation in OpenFOAM [14], employing the finite volume method and a fully time-implicit solution procedure [18]. For the LES case, the Smagorinsky model is used to model the unresolved or subgrid-scale (SGS) momentum and scalar fluxes [1]. A turbulent flame-speed closure (TFC-class) combustion model is used there to account for flame-turbulence interaction, coupled with a technique of tabulated chemistry to include chemical species [19, 20]. For the DNS case, the reaction zones and the turbulent flow are fully resolved, therefore, turbulence and combustion models are not required in this case. An in-house developed solver in OpenFOAM for calculating detailed molecular transport and chemical reactions is used in this case [21–26]. The methane/air combustion is described by a reaction mechanism with 17 species and 58 elementary reactions, together with the mixture-averaged model for transport coefficients.

The SIT generator BC is applied at the inlet boundary for both cases. The turbulence parameters ($Tu = 15\,\%$ and $L_{x,y,z} = 8$ mm) for the LES case have been set according to measured homogeneous turbulence generated by a matrix within the nozzle [17]. In the DNS case, the bulk flow velocity and the turbulence properties at the inlet are adjusted to keep the flame front within the computational domain. Comparing both LES and DNS cases, the LES method on a coarse grid resolves the flame front only down to the cut-off scale, whereas the reacting flow is fully resolved by the DNS method on a sufficiently fine grid. Nonetheless, LES is suitable for dimensionally large-scale simulations. On the contrary, DNS allows accessing more realistic details of the turbulent flame, which, however, is restricted to small domains due to the large computational cost.

4.2 LES of a Turbulent Jet Flame

On the right of Fig. 8, the instantaneous streamwise velocity component $\widetilde{u}$ at the inlet BC calculated by the inflow generator is illustrated, which forms unsteady, non-uniformly distributed flow fluctuations. The influence of the inflow turbulence can also be identified by the contours of $\widetilde{u}$ (left of Fig. 8) and vorticity $\omega = |\nabla \times \widetilde{\mathbf{u}}|$ (middle of Fig. 8) on a meridian cutting-plane passing through the symmetry axis. The flame surfaces are depicted by the solid lines, which are strongly corrugated and torn by the intense turbulence. The turbulent fluctuations are attenuated by passing through the flame surface due to the increased fluid viscosity and thermal expansion. The hot products mix thereafter with incoming cold air which again enhances the turbulence. The temperature increases rapidly in the reaction zone and decreases then

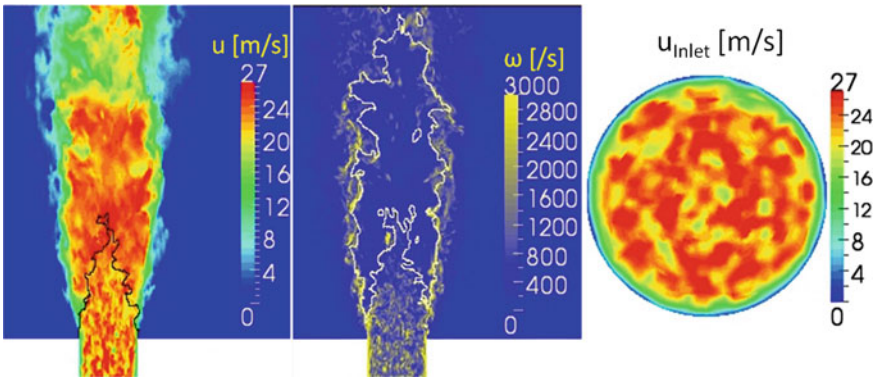


Fig. 8 Instantaneous contours of streamwise velocity u (left) and vorticity ω (middle) on a slice passing through the centerline axis; right subplot shows contours of u generated by the turbulence-generator BC at inlet

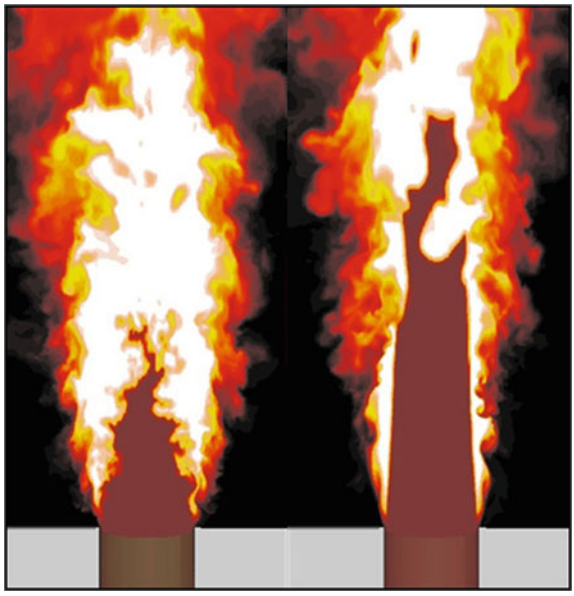


Fig. 9 Comparison of temperature contours with a laminar (left) and turbulent (right) inflow BC

by mixing with the ambient air, which is shown on the left of Fig. 9 in the contour plot of instantaneous temperature.

It is important to note that the precise setting of the turbulence conditions at the inlet boundary plays a decisive role in the current case. Enhanced turbulence leads to a higher turbulent flame speed or an increased overall burning rate. To demonstrate the importance of this effect, another LES simulation is conducted for the considered flame configuration by using the same numerical conditions, however without the inflow generator. As shown in Fig. 9 on the right, LES with a steady-state (laminar) inflow velocity exhibits a rather undisturbed flame front, which is attributed to the missing turbulent fluctuations from the inflow. As a result, it predicts a significantly longer flame than the previous LES with a turbulent inflow. In the latter case, the intense turbulent fluctuations corrugate and wrinkle the flame front, leading to an enhanced flame-turbulence interaction and higher burning rate, so that the flame front propagates more strongly towards the fresh gas mixture. Although not shown here, the simulation results with the laminar inflow do not agree with the measured time-mean and time-rms (root mean square) values for the flow and chemical scalar fields downstream of the burner.

4.3 DNS of a Turbulent Freely Propagating Flame

Figure 10 depicts the flame front calculated by DNS, which is identified by the isotherm of $T \approx 1500$ K. A strongly wrinkled flame surface caused by the turbulent

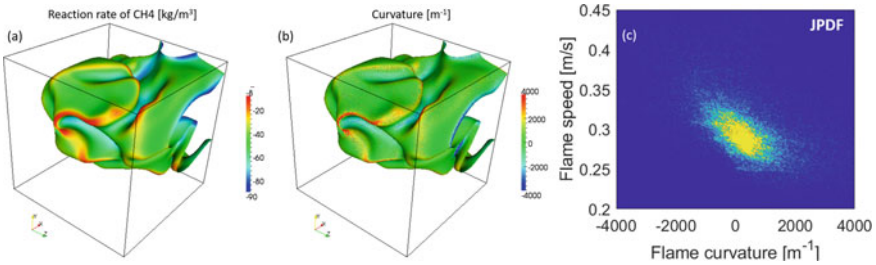


Fig. 10 Instantaneous flame surface contoured by reaction rate of CH_4 **a** and its curvature **b**, and joint probability density function (PDF) between flame speed and flame curvature **c**

flow can be detected. Note that the overall dimension and resolution used for this DNS case is 1 cm^3 and 25 μm. Therefore, the flame structures shown in Fig. 10 cannot be reproduced by LES using grid resolutions generally in the range of $\mathcal{O}(0.1\,mm)$. In this way, the DNS allows an in-depth analysis of the combustion mechanism depending on the flame wrinkling due to turbulent flow. For instance, a strong correlation between the reaction rate of CH_4 and flame surface curvature is clearly found in Fig. 10. This behaviour is further illustrated in Fig. 10 on the right by the joint probability density function (JPDF) between the flame speed and flame curvature, where the flame speed is calculated through integration of the reaction rate of CH_4 along the normals at each element of the flame surface. The flame speed decreases with increasing flame curvature, which is also confirmed by previous experimental and numerical works [27]. The scattering of these quantities is caused by multi-scale interactions of the flame with the turbulent flow both in space and time. The results may be used to develop advanced concepts for turbulent combustion modelling. The study of flame-turbulence interaction considered in this DNS case has been accomplished employing the implemented BC, confirming its validity and usefulness.

5 Conclusion

The implementation, validation and application of a Synthetic Turbulent Inflow (STI) boundary condition (BC) in the framework of OpenFOAM are presented in this work. The approach can generate a realistic turbulent flow field at a given inlet plane, creating temporally and spatially correlated velocity fields. Therefore, it is suited for LES/DNS applications of turbulent flows. The implementation of the BC in OpenFOAM has been described in a detailed way. The importance of such turbulent inflow BC is further demonstrated through LES and DNS combustion simulations, the latter using 8192 CPU cores on Hazel Hen. The proposed implementation of the BC is easier to use compared with the native implementation in OpenFOAM due to a smaller set of necessary input parameters. In contrast to the original concept, the computational efficiently is greatly increased by using coarse virtual grids for the

generation of the inlet boundary velocity field. As a side-effect, this also enables the use of arbitrarily shaped inlets in the simulation.

A performance analysis on the Hawk supercomputer at HLRS demonstrates that the OpenFOAM simulation is memory bound by looking at the intra-node and inter-node performance metrics. Because of this, the highest efficiency is observed when only half the available physical CPU cores per node are utilized. This is because compared to the previous Intel-based cluster Hazel Hen, the memory bandwidth per CPU core is reduced and the L3 cache is shared among a core complex (CCX) on the AMD CPUs of Hawk. The parallel scaling is super-linear up to about 8 000 computational cells per MPI rank on Hawk, beyond which sub-linear scaling is obtained. This is consistent with scaling results obtained at the previous system Hazel Hen. Another problem addressed by the new implementation is that OpenFOAM's native turbulence generator shows inconsistencies in parallel simulations. Because of this, new interpolation routines have been implemented to achieve fully consistent results for large-scale parallel simulations. In conclusion, the new inlet turbulence generator is well suited for large-scale simulations using thousands of CPU cores.

Acknowledgments The authors gratefully acknowledge the financial support by the Helmholtz Association of German Research Centers (HGF), within the research field Energy, Material and Resources, Topic 4 Gasification (34.14.02). This work utilized computing resources provided by the High Performance Computing Center Stuttgart (HLRS) at the University of Stuttgart and the Steinbuch Centre for Computing (SCC) at the Karlsruhe Institute of Technology.

References

1. J. Fröhlich, *Large Eddy Simulation Turbulenter Strömungen*, vol. 1 (Springer, Heidelberg, 2006)
2. M. Klein, A. Sadiki, J. Janicka, A digital filter based generation of inflow data for spatially developing direct numerical or large eddy simulations. J. Comput. Phys. **186**(2), 652–665 (2003)
3. M. Baba-Ahmadi, G. Tabor, Inlet conditions for les using mapping and feedback control. Comput. Fluids **38**(6), 1299–1311 (2009)
4. F. Bazdidi-Tehrani, M. Kiamansouri, M. Jadidi, Inflow turbulence generation techniques for large eddy simulation of flow and dispersion around a model building in a turbulent atmospheric boundary layer. J. Build. Perform. Simul. **9**(6), 680–698 (2016)
5. Z. Rana, B. Thornber, D. Drikakis, On the importance of generating accurate turbulent boundary condition for unsteady simulations. J. Turbul. **12**(12), N35 (2011). Article no. N95. https://www.tandfonline.com/doi/abs/10.1080/14685248.2011.613836
6. F.C.C. Galeazzo, G. Donnert, P. Habisreuther, N. Zarzalis, R.J. Valdes, W. Krebs, Measurement and simulation of turbulent mixing in a jet in crossflow. J. Eng. Gas Turb. Power **133**(6), 10 p. (2011). Article no. 061504. https://asmedigitalcollection.asme.org/gasturbinespower/article-abstract/133/6/061504/407543/Measurement-and-Simulation-of-Turbulent-Mixing-in
7. F. Zhang, P. Habisreuther, M. Hettel, H. Bockhorn, Numerical computation of combustion induced noise using compressible les and hybrid CFD/CAA methods. Acta Acust. Acust. **98**(1), 120–134 (2012)
8. F.C.C. Galeazzo, G. Donnert, C. Cárdenas, J. Sedlmaier, P. Habisreuther, N. Zarzalis, C. Beck, W. Krebs, Computational modeling of turbulent mixing in a jet in crossflow. Int. J. Heat Fluid Flow **41**, 55–65 (2013)

9. F. Zhang, T. Zirwes, H. Nawroth, P. Habisreuther, H. Bockhorn, C.O. Paschereit, Combustion-generated noise: an environment-related issue for future combustion systems. Energ. Technol. **5**(7), 1045–1054 (2017)
10. F. Zhang, T. Zirwes, P. Habisreuther, H. Bockhorn, D. Trimis, H. Nawroth, C.O. Paschereit, Impact of combustion modeling on the spectral response of heat release in LES. Combust. Sci. Technol. **191**(9), 1520–1540 (2019)
11. T. Zirwes, F. Zhang, P. Habisreuther, H. Bockhorn, D. Trimis, Large-scale quasi-DNS of mixed-mode turbulent combustion. PAMM **19**(1), e201900420 (2019)
12. A. Kempf, S. Wysocki, M. Pettit, An efficient, parallel low-storage implementation of klein's turbulence generator for LES and DNS. Comput. Fluids **60**, 58–60 (2012)
13. L. Di Mare, M. Klein, W. Jones, J. Janicka, Synthetic turbulence inflow conditions for large-eddy simulation. Phys. Fluids **18**(2), 025107 (2006)
14. H. Weller, G. Tabor, H. Jasak, C. Fureby, OpenFOAM, openCFD ltd. (2017). https://openfoam.org
15. HPE Apollo (Hawk). High Performance Computing Center Stuttgart (2020). www.hlrs.de/systems/hpe-apollo-hawk
16. T. Zirwes, F. Zhang, J. Denev, P. Habisreuther, H. Bockhorn, D. Trimis, Enhancing OpenFOAM's performance on HPC systems, in *High Performance Computing in Science and Engineering '19*. ed. by W. Nagel, D. Kröner, M. Resch (Springer, Heidelberg, 2019)
17. M. Zajadatz, M. Hettel, W. Leuckel, Burning velocity of high-turbulence natural gas flames for gas turbine application, in *International Gas Research Conference*, vol. 5 (Government Institutes Inc., 1998), pp. 793–803
18. J.H. Ferziger, M. Perić, R.L. Street, *Computational Methods for Fluid Dynamics*, vol. 3 (Springer, Heidelberg, 2002)
19. F. Zhang, P. Habisreuther, M. Hettel, H. Bockhorn, Modelling of a premixed swirl-stabilized flame using a turbulent flame speed closure model in LES. Flow Turbul. Combust. **82**(4), 537–551 (2009)
20. F. Zhang, P. Habisreuther, H. Bockhorn, H. Nawroth, C. Paschereit, On prediction of combustion generated noise with the turbulent heat release rate. Acta Acust. Acust. **99**(6), 940–951 (2013)
21. F. Zhang, H. Bonart, T. Zirwes, P. Habisreuther, H. Bockhorn, N. Zarzalis, Direct numerical simulation of chemically reacting flows with the public domain code openfoam, in *High Performance Computing in Science and Engineering 2014* (Springer, 2015), pp. 221–236
22. T. Zirwes, F. Zhang, J. A. Denev, P. Habisreuther, H. Bockhorn, Automated code generation for maximizing performance of detailed chemistry calculations in openfoam, in *High Performance Computing in Science and Engineering 2017* (Springer, 2018), pp. 189–204
23. T. Zirwes, F. Zhang, J. Denev, P. Habisreuther, H. Bockhorn, Improved vectorization for efficient chemistry computations in OpenFOAM for large scale combustion simulations, in *High Performance Computing in Science and Engineering '18*. ed. by W. Nagel, D. Kröner, M. Resch (Springer, Berlin Heidelberg, 2018)
24. T. Zirwes, F. Zhang, P. Habisreuther, M. Hansinger, H. Bockhorn, M. Pfitzner, D. Trimis, Quasi-DNS dataset of a piloted flame with inhomogeneous inlet conditions. Flow Turbul. Combust. **104**, 997–1027 (2019)
25. F. Zhang, T. Baust, T. Zirwes, J. Denev, P. Habisreuther, N. Zarzalis, H. Bockhorn, Impact of infinite thin flame approach on the evaluation of flame speed using spherically expanding flames. Energ. Technol. **5**(7), 1055–1063 (2017)
26. T. Zirwes, F. Zhang, T. Häber, H. Bockhorn, Ignition of combustible mixtures by hot particles at varying relative speeds. Combust. Sci. Tech. **191**, 178–195 (2019)
27. F. Zhang, T. Zirwes, P. Habisreuther, H. Bockhorn, Effect of unsteady stretching on the flame local dynamics. Combust. Flame **175**, 170–179 (2017)

Implementation of Lagrangian Surface Tracking for High Performance Computing

Thorsten Zirwes, Feichi Zhang, Jordan A. Denev, Peter Habisreuther, Henning Bockhorn, and Dimosthenis Trimis

Abstract In almost all technically relevant combustion applications, flames occur in turbulent flows. The interaction of turbulent flows with flames is still not fully understood due to the large range of time and length scales which govern combustion processes. One method of studying this interaction is by tracking thermo-physical trajectories of material points on flame surfaces. These trajectories give insight into the local flame dynamics and help to understand the influence of turbulence on flame properties. In this work, a Lagrangian tracking algorithm is presented which performs the tracking of material points on iso-surfaces. Because this tracking method is used in large-scale direct numerical simulations of combustion processes, the focus of the implementation lies on performance. By tracking the position of the Lagrangian particles in barycentric coordinates, efficient algorithms for spatial interpolation and the intersection of particle trajectories with iso-surfaces can be utilized. The code is written in a general way and not restricted to reacting flows but can be used to track any iso-surface. Additionally, the algorithm works by decomposing the computational cells into tetrahedra. This allows the tracking method to work on unstructured meshes with arbitrary cell shapes. The tracking method is implemented in OpenFOAM and applied to the direct numerical simulation of a 3D turbulent flame. The simulations are conducted with a custom solver which makes use of automatically generated, highly optimized code for the computation of chemical reaction rates, which performs up to 20 times faster than OpenFOAM's implementation. For the turbulent flame, applying the tracking method increases simulation times by less than 5 %, so that the current implementation is well suited to be used during large scale simulations.

T. Zirwes (✉) · J. A. Denev
Steinbuch Centre for Computing, Karlsruhe Institute of Technology,
Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany
e-mail: thorsten.zirwes@kit.edu

T. Zirwes · F. Zhang · P. Habisreuther · H. Bockhorn · D. Trimis
Engler-Bunte Institute (Chair of Combustion Technology), Karlsruhe Institute of Technology,
Engler-Bunte-Ring 7, 76131 Karlsruhe, Germany
e-mail: feichi.zhang@kit.edu

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_15

Keywords Lagrangian tracking · Direct numerical simulation · Reacting flows · OpenFOAM · High performance computing

1 Introduction

Due to the current climate goals and increasingly stringent emission limits, there is a strong need for novel and more efficient energy devices. While the world's energy consumption increases, carbon emissions have to decrease drastically. At the same time, combustion will provide most of the energy for the global primary energy consumption, transport sector and power generation at least until the year 2040 [1]. It is therefore mandatory to study combustion processes in greater detail to enable the development of more efficient combustion devices.

One powerful tool for studying combustion is direct numerical simulation (DNS). In this simulation technique, all relevant time and length scales of both the flow and the flame have to be resolved. However, these time and length scales can span ten orders of magnitudes. For example, the residence time of the flow inside a combustion device may be in the order of seconds while the fastest chemical reactions take place i the order of 10^{-10} s. Likewise, the physical dimensions of a combustion device might be of the order of meters, while the thickness of the reaction zone of the flame or the smallest eddies in a turbulent flow field might be in the order of micrometers.

Apart from this discrepancy in length and time scales, the combustion chemistry itself constitutes a challenge for direct numerical simulation. During the combustion, many intermediate chemical species, such as radicals, are produced which interact through a large number of chemical reactions. For complex fuels, many hundred chemical species and thousands of chemical reactions have to be taken into account if a detailed result of the flame dynamics is desired [2]. It is therefore evident that the direct numerical simulation of turbulent combustion is only possible on modern supercomputers. Even then, simulations are mostly limited to flames of laboratory scale.

Although DNS is one of the best tools to study combustion processes, it generates large amounts of data. This means that it is unfeasible to store the full three-dimensional information of the flame and flow field at a high time-frequency. Because of this, it is valuable to perform analysis of the flame-dynamics in-situ to decrease post-processing times and avoid the storage of excessive amounts of data.

A promising approach for studying the interaction of the turbulent flow with flames is the flame particle tracking (FPT) method [3]. In this method, an iso-surface of e.g. temperature or a species concentration is chosen, which represents the flame front. Special mathematical points on that iso-surface, so called material points, are tracked over time by using virtual Lagrangian particles, called flame particles (FP). These particles are massless and have no size and do not interact with the flow field. They merely serve as markers for the surface tracking. Because these particles co-move with the flame, their trajectories allow to analyze the local flame dynamics in a time resolved manner. Another useful property of these flame particles is, that their

movement is directly linked to the flame stretch [4], which is an important quantity in combustion technology.

Although the method of flame particle tracking has been used before in the literature [3, 5–8], is has always been done as a post-processing step, mostly in Matlab. This required the aforementioned storage of large amounts of data and also strongly limited the utilization of parallel hardware. Because of this, the number of concurrently tracked flame particles was strongly limited.

In this work, a fully parallelized implementation [4] of the flame particle tracking method is presented, which can be used directly during the simulation. Another advantage of this implementation is the use of barycentric coordinates. Instead of tracking the position of the flame particles in Cartesian coordinates, which is usually a three-dimensional vector, their position is tracked in barycentric coordinates. The computational cells are decomposed into tetrahedra so that each barycentric coordinate is a four-dimensional vector. Each component is the normalized weight of one of the four points of the tetrahedron of which the flame particle is inside. This has several advantages:

- the method can be used with arbitrary cell shapes because any polyhedron can be decomposed uniquely into tetrahedra
- interpolation in barycentric coordinates is computationally very efficient
- geometrical operations like finding the intersection of particle trajectories with the iso-surface are computationally efficient as well.

In Sect. 2, the mathematical basics of the flame particle tracking method are explained and the details of the implementation are provided. Section 3 gives a brief overview of the in-house [9] OpenFOAM [10] combustion solver and compares the computational performance with OpenFOAM's default solvers. Section 4 presents the numerical setups of the flame simulations and shows some preliminary results from the flame particle tracking method. Finally, Sect. 5 discusses the performance of the Lagrangian tracking implementation in terms of computational overhead, memory consumption and parallel scaling.

2 Tracking Algorithm

2.1 Mathematical Basics

As explained in the introduction, the goal of the flame particle tracking (FPT) method is to track material points on iso-surfaces with Lagrangian particles (*flame particles*). The movement of any iso-surface, that is a material surface, consists of two components: convection with the fluid flow velocity $\vec{u}$ and an additional intrinsic velocity. In the case of flames, this intrinsic velocity is called the burning velocity, or more accurately the displacement speed s_d of the flame or iso-surface. Additionally, a

flame only propagates normal to itself. Therefore, the movement velocity $vvecw$ of a material point on the flame iso-surface is given by

$$\vec{w} = \vec{u} + s_d \vec{n} \tag{1}$$

$\vec{n}$ is the unit normal vector of the iso-surface. If Φ is the iso-scalar, which might be temperature or the concentration of a chemical species, and $\Phi = \Phi_{\text{iso}}$ defines the iso-surface, the unit normal vector can be computed by:

$$\vec{n} \equiv \pm \left(\frac{\nabla \Phi}{|\nabla \Phi|} \right)_{\Phi = \Phi_{\text{iso}}} \tag{2}$$

The sign depends on the convention, e.g. if a reactant or product specie is chosen [4]. The displacement speed s_d can be computed from

$$s_d = - \left(\frac{1}{|\nabla \Phi|} \frac{\mathrm{D}\Phi}{\mathrm{D}t} \right)_{\Phi = \Phi_{\text{iso}}} \tag{3}$$

which is an exact expression. $\mathrm{D}/\mathrm{D}t$ is the substantial derivative. If a balance equation for the quantity Φ is solved during the simulation, the evaluation of Eq. (3) is computationally cheap because the time derivative is already known. This forms the mathematical basis for computing the movement speed $\vec{w}$ of material points on the iso-surface and the flame particles.

2.2 Tracking Algorithm

After seeding the flame particles on an initial iso-surface, the implemented tracking algorithm is comprised of the following steps:

1. compute movement velocity $\vec{w}$ and interpolate it at each FP position
2. using $\vec{w}$, compute the position of each flame particle at the next time step
3. move the flame particles towards the new destination
4. if the particle crosses a domain boundary between different MPI processes, move the particle to the neighboring process
5. after the tracking, correct the particle position to make sure that it is located exactly on the iso-surface.

Figure 1 demonstrates steps 1–3. It shows two computational cells that are cut by the iso-surface. The blue spheres are the cell centers, which become important in the last step of the algorithm. The flame particle (FP) (red sphere) in the cell on the right was located on the iso-surface at $t = t_0$. In the new time step $t = t_1$, the flame and therewith the iso-surface has moved. Therefore, in a first step, the movement speed $\vec{w}$ from Eq. (1) is computed and interpolated to the particle position. In the most simple case, the new position of the flame particle at $t = t_1$ can be computed from

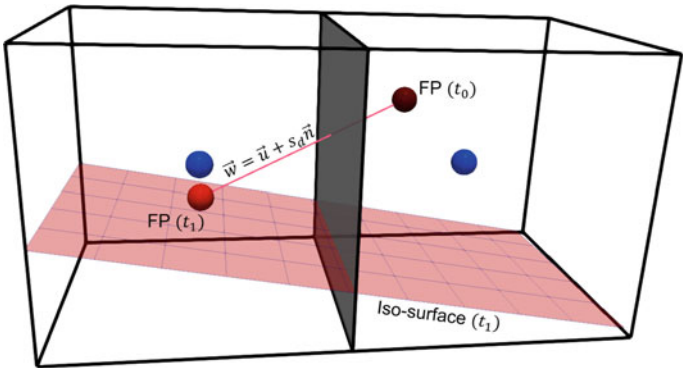


Fig. 1 Example for the tracking. Depicted are two hexahedral cells in the computational mesh, where the cell centers are illustrated by blue spheres. The red sphere in the cell on the right is a flame particle at a previous time step ($t = t_0$) which is moved to its new position in the left hand side cell on the iso-surface at $t = t_1$. In this case, the particle has to cross the face between the cells, which might be a sub-domain boundary between different MPI processes.

$$\vec{x}_{\mathrm{FP}}(t_1) \approx \vec{x}_{\mathrm{FP}}(t_0) + \vec{w}\Delta t \tag{4}$$

where Δt is the simulation time step. The accuracy of this can be further improved by using a predictor-corrector approach [3]. In the scenario depicted in Fig. 1, the flame particle has to move from its original cell into another cell. The cell face in-between these two cells (shown as gray area) might actually be the boundary between different parallel processes.

Since this method is implemented into OpenFOAM [11], the existing infrastructure of the Lagrangian framework can be used to handle the parallelization: In OpenFOAM, the computational mesh is decomposed into different sub-domains and assigned to different MPI processes. Therefore, every time a Lagrangian particle hits a cell face, it is checked if this cell face is a process boundary. If it is, the tracking is stopped. Then, all particles that have hit a process boundary are deleted on the current MPI process and sent to the MPI process which owns the neighbor cell. This is repeated until all particles have reached either their destination or a physical boundary.

After the tracking, the particles have reached a point that is very close to the iso-surface. However, due to numerical errors, they are most likely not located exactly on the iso-surface. This is not a problem immediately, but over the course of many time steps, the numerical errors accumulate and the particles drift away from the iso-surface. Because of this, the last step of the tracking is a correction procedure which ensures that the particles end up exactly on the iso-surface.

The correction procedure starts by identifying which tetrahedron of the decomposed cell the flame particle is located in, in order to utilize barycentric coordinates. Figure 2 illustrates this situation: The goal is to find the shortest distance ΔC between the flame particle (red sphere) and the iso-surface. In OpenFOAM, the position of the

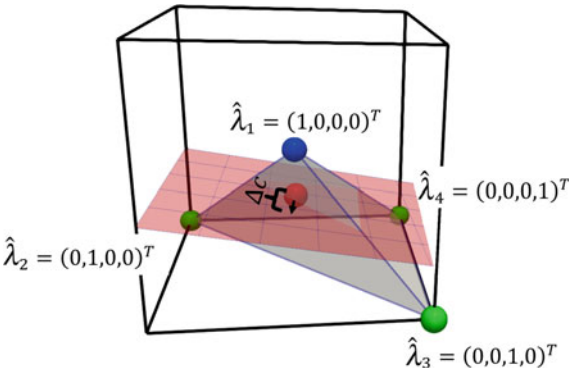


Fig. 2 After moving the flame particle (red sphere) to its new position, the shortest distance between the particle and the iso-surface ΔC is computed to correct the particle position. This is done using barycentric coordinates relative to the cell tetrahedron the flame particle is located in, comprised of three corner points of the cell and the cell center.

particle $\hat{\lambda}_{\mathrm{FP}}$ is stored in barycentric coordinates relative to the current tetrahedron. The vector $\hat{\lambda}_{\mathrm{FP}}$ is a four-dimensional vector, where each component represents how near the position is to each corner of the tetrahedron. We choose the notation where $\vec{v}$ denotes a 3×1 vector with Cartesian coordinates and $\hat{v}$ denotes a 4×1 vector with the barycentric coordinates. When decomposing the cell into tetrahedra, one of the corners is always formed by the cell center, and the other three corner points of the tetrahedron are formed by three corner points of the cell. This is true for any polyhedral cell shape, e.g. a hexahedral cell is split into 12 tetrahedra (two tetrahedra per cell face), as shown by one of the twelve tetrahedra in Fig. 2. For example, the barycentric coordinate of the cell center in the tetrahedron is $\hat{\lambda}_1 = (1, 0, 0, 0)^T$, because it has the shortest possible distance to the first corner point of the tetrahedron (itself) and the maximum distance to the other corner points. Therefore, all corner points are identified by a $\hat{\lambda}$, where one component is unity and all other components are zero. Another property of any barycentric coordinate $\hat{\lambda} = (\lambda_1, \lambda_2, \lambda_3, \lambda_4)^T$ is that

$$\sum_{i=1}^{4} \lambda_i = 1, \ \lambda_i \geq 0 \tag{5}$$

For any barycentric coordinate $\hat{\lambda}$ inside that tetrahedron, the value of Φ can be found by interpolation:

$$\Phi(\hat{\lambda}) = \sum_{i=1}^{4} \lambda_i \Phi_i \tag{6}$$

Here, Φ_i, is the value of Φ at the corner point i of the tetrahedron. Because of this, any quantity can be interpolation computationally very efficiently inside that tetrahedron, e.g. at the position of the flame particle. Based on Eq. (6), the iso-surface that cuts this tetrahedron can be parameterized by

$$\Phi_{\text{iso}} = \sum_{i=1}^{4} \lambda_i \Phi_i \tag{7}$$

Any point $\hat{\lambda}$ that fulfills this equation lies on the iso-surface. In previous works [3], the intersection of the particle trajectory with the iso-surface was found by reconstructing the whole iso-surface and performing ray tracing. This can be avoided entirely when using the barycentric coordinate approach. The shortest distance d between a point $\vec{p}$ and a plane can be computed from

$$d = (\vec{p} - \vec{q}) \cdot \vec{n} = \vec{p} \cdot \vec{n} - \vec{q} \cdot \vec{n} \tag{8}$$

where $\vec{q}$ is an arbitrary point that lies on the plane. In this case, the point $\vec{p}$ is the position of the flame particle $\vec{p}_{\text{FP}}$. However, this position is stored in barycentric coordinates in OpenFOAM $\hat{\lambda}_{\text{FP}}$. The conversion from barycentric to Cartesian coordinates is done by

$$\vec{p}(\hat{\lambda}) = \lambda_1 \vec{x}_1 + \lambda_2 \vec{x}_2 + \lambda_3 \vec{x}_3 + \lambda_4 \vec{x}_4 = \mathbf{T}\,\hat{\lambda}, \qquad \mathbf{T} = (\vec{x}_1\ \vec{x}_2\ \vec{x}_3\ \vec{x}_4) \tag{9}$$

$\vec{x}_i$ are the coordinates of the four corner points of the tetrahedron in Cartesian coordinates. $\mathbf{T}$ is a 3×4 matrix whose columns are formed by $\vec{x}_i$. Combining Eq. (8) and Eq. (9) to find the shortest distance between the flame particle and the iso-surface yields $d = \left(\mathbf{T}\,\hat{\lambda}_{\text{FP}}\right) \cdot \vec{n} - \vec{q} \cdot \vec{n}$. The only unknown is the point $\vec{q}$. A very efficient way of finding a point $\vec{q}$ that lies on the iso-surface is identifying any edge of the tetrahedron that is cut by the iso-surface. Any point on the edge expressed in barycentric coordinates will have two components that are zero, which correspond to the two points that are not part of that edge. For example, assume the edge from the corner point 1 to the corner point 3 is cut by the iso-surface (see e.g. Fig. 2). Then the point on that edge which also lies on the iso-surface must have the form $\hat{\lambda}_{\text{edge}} = (\lambda_1,\ 0,\ \lambda_3,\ 0)$. Because of Eq. (7), this point also has to fulfill $\Phi_{\text{iso}} = \sum_{i=1}^{4} \lambda_i \Phi_i$, which for this example reduces to

$$\Phi_{\text{iso}} = \lambda_1 \Phi_1 + \lambda_3 \Phi_3 \tag{10}$$

And because of Eq. (5), the sum of the components of the barycentric coordinate vector has to be unity. In this example, this yields

$$\lambda_1 + \lambda_3 = 1,\ \lambda_1,\ \lambda_3 \geq 0 \tag{11}$$

The two unknowns λ_1 and λ_3 can be found with help of the two Eqs. (10) and (11). In this way, a point on the iso-surface can be found efficiently by choosing an edge and finding the point of that edge that is cut be iso-surface $\hat{\lambda}_{\text{edge}}$, which has the special property that two of the four components are zero. With Eq. (9), the Cartesian coordinate of that point is found by $\vec{q} = \mathbf{T}\,\hat{\lambda}_{\text{edge}}$. Substituting this into Eq. (8) finally yields the equation for efficiently finding the shortest distance between the flame particle and the iso-surface (see also [4]):

$$\Delta C = (\mathbf{T}\,\hat{\lambda}_{\mathrm{FP}}) \cdot \vec{n} - (\mathbf{T}\,\hat{\lambda}_{\mathrm{edge}}) \cdot \vec{n} = \left(\mathbf{T}\left(\hat{\lambda}_{\mathrm{FP}} - \hat{\lambda}_{\mathrm{edge}}\right)\right) \cdot \vec{n} \tag{12}$$

The normal vector $\vec{n}$ can be computed from Eq. (2). After correcting the flame particle position by $\Delta C\,\vec{n}$, it exactly lies on the iso-surface.

It can happen that the intersection point of the flame particle and the iso-surface lies in another tetrahedron. If this is the case, the flame particle is iteratively moved toward the iso-surface until it reaches the correct tetrahedron. The distance Δx between the flame particle and the iso-surface can be found by using the approximation $\partial\Phi/\partial n \approx (\Phi_{\mathrm{iso}} - \Phi_{\mathrm{FP}})/\Delta x \rightarrow \Delta x \approx (\Phi_{\mathrm{iso}} - \Phi_{\mathrm{FP}})/(\partial\Phi/\partial n)$. The correction is done in the direction of the local normal vector $\Delta x\,\vec{n}$. Using the definition of the directional derivative $\partial\Phi/\partial n \equiv \nabla\Phi \cdot \vec{n}$ together with the definition of $\vec{n}$ from Eq. (2) therefore yields

$$\Delta x\,\vec{n} \approx \frac{\Phi_{\mathrm{iso}} - \Phi_{\mathrm{FP}}}{\frac{\partial\Phi}{\partial n}}\vec{n} = \frac{\Phi_{\mathrm{iso}} - \Phi_{\mathrm{FP}}}{\nabla\Phi_{\mathrm{FP}} \cdot \vec{n}}\vec{n} = \frac{\Phi_{\mathrm{iso}} - \Phi_{\mathrm{FP}}}{\nabla\Phi_{\mathrm{FP}} \cdot \frac{\nabla\Phi_{\mathrm{FP}}}{|\nabla\Phi_{\mathrm{FP}}|}} \frac{\nabla\Phi_{\mathrm{FP}}}{|\nabla\Phi_{\mathrm{FP}}|} \tag{13}$$

$$= \epsilon(\Phi_{\mathrm{iso}} - \Phi_{\mathrm{FP}}) \frac{\nabla\Phi_{\mathrm{FP}}}{|\nabla\Phi_{\mathrm{FP}}|^2} \tag{14}$$

Φ_{FP} is the value of Φ interpolated at the particle position and $\nabla\Phi_{\mathrm{FP}}$ its gradient at the particle position. ϵ in the last step is introduced as an underrelaxation factor, which is usually set to 0.9. In this way, computing the exact location on the iso-surface can be done with little computational effort and without the need of reconstructing large portions of the iso-surface or performing ray tracing.

3 Reacting Flow Solver for Large Scale Simulation

The flame particle tracking method is implemented in the framework of OpenFOAM. In order to perform direct numerical simulation of turbulent flames, a custom solver for reacting flows has been created, which improves several aspects of OpenFOAM necessary for combustion DNS and has been validated by many different flame configurations [9, 12–29].

The second major difference to OpenFOAM's standard combustion solvers is, that the computational performance is drastically improved. As mentioned in the introduction, the combustion process comprises many chemical species and a large number of chemical reactions. The chemical reaction rates depend exponentially on temperature so that the underlying equations are numerically very stiff. Therefore, the custom solver is coupled to a highly-optimized solver for stiff ordinary differential equations called CVODE by Sundials [31]. In addition to this, a converter tool has been developed which generates C++ source code that is optimized for different chemical systems. More details about this can be found in [15, 16]. Together, these two developments increase the performance of the chemistry calculations by on order of magnitude. For example, in a simple test-case for 0D auto-ignition of hydrogen

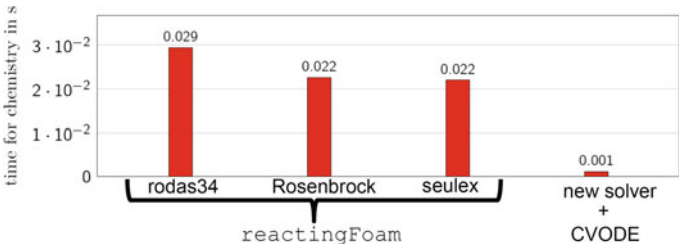


Fig. 3 Comparison of the time spent on chemistry computations for the 0D auto-ignition of hydrogen in air [30] of OpenFOAM's standard solver `reactingFoam` using three different ODE solvers (rodas34, Rosenbrock and seulex) with the new solver, which uses Sundials CVODE. A speedup of about 20 is achieved between the new solver and the best settings for the standard OpenFOAM solver without loss of accuracy.

in air, the computation of chemical reaction rates is 20 times faster in the new solver compared to the best configuration in OpenFOAM (Fig. 3).

4 Turbulent Flame Case

The new flame particle tracking algorithm is applied to the direct numerical simulation of a turbulent flame. A typical model flame is the so called *synthetic flame in a box*, where a flame is placed in a rectangular domain. A turbulence generator at the inlet creates a turbulent flow, which interacts with the originally planar flame front. The computational domain consists of 64 million cells and the mesh is equidistantly spaced with a total side length of 1 cm. The reaction mechanism by Li et al. [30] is used to described the oxidation of hydrogen, which comprises 9 chemical species and 21 chemical reactions. The time step is set to 5×10^{-7} s, which corresponds to a Courant number of 0.2. The governing equations are solved with the finite volume method. Temporal derivatives are discretized with a second order backward Euler scheme, and a fourth order interpolation scheme is chosen for spatial discretization. This case serves as a test case for the flame particle tracking. Figure 4 shows results of a simulation where the flame particles are seeded onto an iso-surface of the hydrogen mass fraction. At the beginning of the simulation (left), one particle is seeded at every cell that is cut by the iso-surface (only a random sub-set of those particles is visualized). After the simulation is run for some time, the particles stay on the iso-surface but change their distribution: Some parts of the flame collect the flame particles while other parts of the flame repel them. Because the flame particles are seeded onto the flame surface only once at the start of the simulation, studying the distribution of the flame particles over time identifies regions on the flame where flame surface area is created (low concentration of flame particles, positive flame stretch) and destroyed (high concentration of flame particles, negative flame stretch).

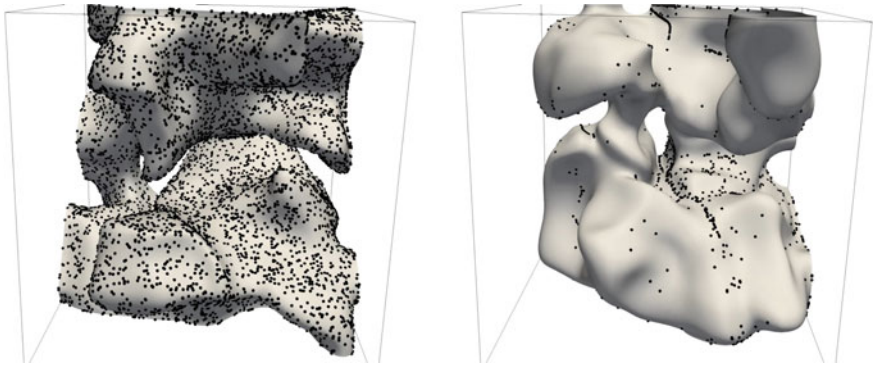


Fig. 4 Iso-surface of the hydrogen mass fraction at the beginning of the simulation (left) and at a later time (right). Black dots show the position of flame particles.

5 Performance

Since the flame particle tracking is done in-situ, it is important that its impact on simulation performance stays minimal. Because direct numerical simulation is only possible on supercomputers, an efficient utilization of the parallel hardware is mandatory. In this section, three aspects of the implementation are discussed: memory overhead, computational overhead and parallel performance. The basis for the performance analysis is the flame in a box from the previous section, but using a methane/air flame instead of a hydrogen/air flame. The number of computational cells in the domain is 64 million. At the beginning of the simulation, one flame particle is seeded onto the iso-surface for every cell that is cut by the iso-surface. This is the common use-case for the flame particle tracking. In this specific case, the total number of flame particles at the start of the simulation is 500 000.

One concern for the simulation is the memory consumption. As modern supercomputers move toward systems with larger numbers of computing cores per node, the available memory per core typically reduces. Therefore, it is important that the flame particle tracking implementation does not add a significant memory overhead.

Because the flame particles are virtual particles, they do not need to store any physical properties like diameter or velocity. Therefore, the memory required to represent a flame particle is very low compared to other Lagrangian particle types. Table 1 shows the memory consumption for one particle.

In total, the size of one particle object (including padding) is usually 128 Bytes. In comparison, the amount of data that has to be stored per cell in the computational mesh is much larger. Just storing the mass fractions of each chemical species (18 in this case) for two time steps (second order time discretization) results in 288 Bytes. In addition to this, information about the mesh topology, transport coefficients for each chemical species and thermo-physical properties like heat capacity have to be stored as well.

Table 1 Memory required to store one flame tracking particle (assuming 32bit integers and 64bit floating points).

Information	Size (Bytes)
Position in barycentric coordinates	32
Information about tetrahedron	24
MPI rank of process	4
Unique ID	4
Coordinate of new position	24
Additional bookkeeping	24

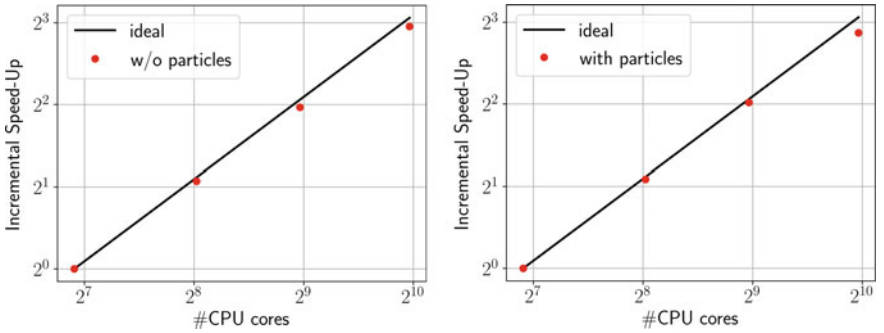


Fig. 5 Incremental speed-up of a simulation without flame particle tracking (left) and with flame particle tracking (right).

Because the flame particles only exist on a 2D (iso-)surface by definition, the number of flame particles is much smaller than the number of cells in the computational mesh (500 000 particles vs. 64 million cells). Therefore, the memory overhead introduced by using the flame particle tracking during the simulation is negligible.

The parallel scaling behavior was tested on the ForHLR 2 cluster at the Steinbuch Centre for Computing at the Karlsruhe Institute of Technology with 120 CPU cores to 1000 CPU cores. Figure 5 shows the scaling.

On the left of Fig. 5, scaling results without Lagrangian particle tracking show that parallel efficiency with 1000 cores is $> 90\,\%$. The scaling results with 500 000 particles on the right are very similar, reaching parallel efficiency of over 90 % too. One consequence of having the particles track the iso-surface is, that they only exist on processes where the flame is located. This technically leads to load imbalances between the processes. However, the load imbalance due to the computation of chemical reaction rates is much more severe and has been addressed in the past [16, 21].

Similar to the memory overhead, the computational overhead per particle is very low. In contrast to other Lagrangian particle types, which require the computation of forces or thermo-chemical effects and utilize sub-time stepping for time integration, the flame particles can directly be moved to the computed destination.

Table 2 Time for one simulation time step with 1000 CPU cores.

Particles	Time per time step (s)
0	14.8
500 000	14.9
5 000 000	15.0
50 000 000	18.5
500 000 000	57.9

Figure 6 shows the average time required to compute one time step with and without particle tracking. On average, the computational overhead in terms of simulation times is about 5 % (Table 2).

Although the case with 500 000 particles constitutes the usual use-case, simulations with much larger number of particles have been conducted as well. Increasing the number of particles by a factor 10 does not affect the overall simulation time. With this amount of particles, the tracking itself is not the limiting factor, but the computation of $\vec{w}$. Even using 100 times as many particles as in a reasonable use-case, the simulation time increases by less than 30 %. Therefore, the implementation is able to handle a large number of particles efficiently on highly parallel systems.

6 Summary

A new flame particle tracking method with focus on high performance computing has been implemented into OpenFOAM. It allows to track the movement of material points on iso-surfaces for studying the local flame dynamics during flame-turbulence interaction. The code is written in a general way and can be used to track any iso-surface, not limited to reactive flows, e.g. to study the correlation of flow velocities

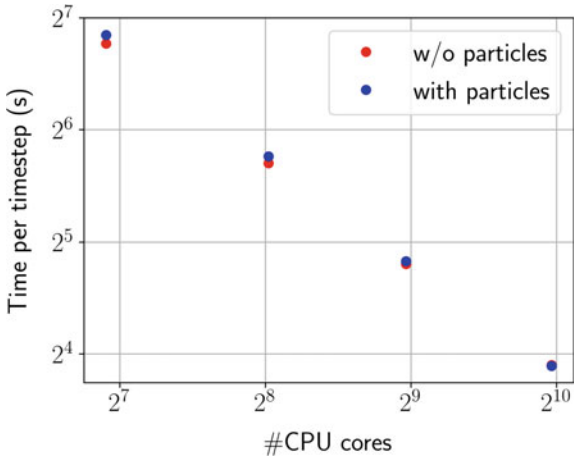


Fig. 6 Time for one simulation time step with and without particle tracking.

and streamline curvature. In contrast to other works from literature, which applied the flame particle tracking method as a post-processing step in serial, the presented implementation is fully parallel. The use of barycentric coordinates enables computationally efficient interpolation of arbitrary quantities to the flame particle's position and allows to find the intersection of the particle trajectory with the iso-surface with an algebraic expression instead of using expensive ray tracing.

A 3D DNS of a turbulent flame serves as a test case for the flame particles. Studying the distribution of flame particles on the flame can identify regions of flame surface creation and destruction. Both the performance and memory overhead of using the implemented particle tracking method are negligible. The parallel scaling of the combustion solver is very good and unaffected by the new tracking method. In conclusion, the newly implemented flame particle tracking method is suited for application in high performance computing and will help to gain a deeper understanding of the complex interplay of turbulent flows and flames.

Acknowledgments This work was supported by the Helmholtz Association of German Research Centres (HGF) through the Research Unit EMR, Topic 4 Gasification (34.14.02) and performed on the national supercomputer Cray XC40 Hazel Hen at the High Performance Computing Center Stuttgart (HLRS) and ForHLR II with the acronym DNSbomb funded by the Ministry of Science, Research and the Arts Baden-Württemberg and DFG ("Deutsche Forschungsgemeinschaft").

References

1. BP energy outlook 2019, (British Petroleum 2019)
2. T. Poinsot, D. Veynante, *Theoretical and Numerical Combustion*. (RT Edwards, Inc. 2005)
3. S. Chaudhuri, Life of flame particles embedded in premixed flames interacting with near isotropic turbulence. Proc. Combust. Inst. **35**(2), 1305–1312 (2015)
4. T. Zirwes, F. Zhang, Y. Wang, P. Habisreuther, J. Denev, Z. Chen, H. Bockhorn, D. Trimis, In-situ flame particle tracking based on barycentric coordinates for studying local flame dynamics in pulsating bunsen flames. in *Proceedings of the Combustion Institute*, vol. 38, (Elsevier 2020)
5. S. Pope, The evolution of surfaces in turbulence. Int. J. Eng. Sci. **26**(5), 445–469 (1988)
6. P. Sripakagorn, S. Mitarai, G. Kosály, H. Pitsch, Extinction and reignition in a diffusion flame: a direct numerical simulation study. J. Fluid Mech. **518**, 231–259 (2004)
7. S. Chaudhuri, Pair dispersion of turbulent premixed flame elements. Phys. Rev. E **91**(2), 021001 (2015)
8. H.L. Dave, A. Mohan, S. Chaudhuri, Genesis and evolution of premixed flames in turbulence. Combust. Flame **196**, 386–399 (2018)
9. T. Zirwes, F. Zhang, P. Habisreuther, M. Hansinger, H. Bockhorn, M. Pfitzner, D. Trimis, "Quasi-DNS dataset of a piloted flame with inhomogeneous inlet conditions," flow. Turb. Combust. **104**, 997–1027 (2019)
10. OpenCFD, *OpenFOAM: The Open Source CFD Toolbox. User Guide Version 1.4, OpenCFD Limited*. Reading UK, Apr. 2007
11. T. Zirwes, F. Zhang, A. Jordan, P. Habisreuther, H. Bockhorn, D. Trimis, "Lagrangian tracking of material surfaces in reacting flows, in *OpenFOAM Workshop*, vol. 15 (2020)
12. F. Zhang, H. Bonart, T. Zirwes, P. Habisreuther, H. Bockhorn, N. Zarzalis, Direct numerical simulation of chemically reacting flows with the public domain code OpenFOAM, in *High Performance Computing in Science and Engineering '14*. ed. by W. Nagel, D. Kröner, M. Resch (Springer, Berlin Heidelberg, 2015), pp. 221–236

13. F. Zhang, T. Zirwes, P. Habisreuther, H. Bockhorn, A DNS analysis of the evaluation of heat release rates from chemiluminescence measurements in turbulent combustion, ed. by W. Nagel, D. Kröner, M. Resch, in *High Performance Computing in Science and Engineering 2016*, (Springer, 2016), pp. 229–243 https://doi.org/10.1007/978-3-319-47066-5_16
14. F. Zhang, T. Zirwes, P. Habisreuther, H. Bockhorn, Effect of unsteady stretching on the flame local dynamics. Combust. Flame **175**, 170–179 (2017)
15. T. Zirwes, F. Zhang, J. Denev, P. Habisreuther, H. Bockhorn, Improved vectorization for efficient chemistry computations in OpenFOAM for large scale combustion simulations, in *High Performance Computing in Science and Engineering '18*. ed. by W. Nagel, D. Kröner, M. Resch (Springer, Berlin Heidelberg, 2018), pp. 209–224
16. T. Zirwes, F. Zhang, J.A. Denev, P. Habisreuther, H. Bockhorn, Automated code generation for maximizing performance of detailed chemistry calculations in OpenFOAM, in *High Performance Computing in Science and Engineering 2017*. (Springer, 2018), pp. 189–204
17. T. Zirwes, Effect of stretch on the burning velocity of laminar and turbulent premixed flames, Master's Thesis, Karlsruhe Institute of Technology (2016)
18. T. Zirwes, F. Zhang, P. Habisreuther, H. Bockhorn, D. Trimis, Large-scale quasi-DNS of mixed-mode turbulent combustion. PAMM **19**(1), e201900420 (2019)
19. F. Zhang, T. Zirwes, P. Habisreuther, H. Bockhorn, D. Trimis, H. Nawroth, C.O. Paschereit, Impact of combustion modeling on the spectral response of heat release in LES. Combust. Sci. Technol. **191**(9), 1520–1540 (2019)
20. T. Zirwes, F. Zhang, T. Häber, H. Bockhorn, Ignition of combustible mixtures by hot particles at varying relative speeds. Combust. Sci. Tech. **191**, 178–195 (2019)
21. T. Zirwes, F. Zhang, P. Habisreuther, J. Denev, H. Bockhorn, D. Trimis, Optimizing load balancing of reacting flow solvers in openfoam for high performance computing, in *Proceedings of 6th ESI OpenFOAM User Conference*, vol. 6 (2018)
22. T. Zirwes, F. Zhang, J. Denev, P. Habisreuther, H. Bockhorn, D. Trimis, Enhancing OpenFOAM's performance on HPC systems, in *High Performance Computing in Science and Engineering '19*. ed. by W. Nagel, D. Kröner, M. Resch (Springer, Berlin Heidelberg, 2019)
23. M. Soysal, M. Berghoff, T. Zirwes, M. Vef, S. Oeste, A. Brinkman, W. Nagel, A. Streit, "Using On-demand File Systems in HPC Environments, in *International Conference on High Performance Computing and Simulation, 6th Special Session on High Performance Computing Benchmarking and Optimization* (2019)
24. T. Zirwes, F. Zhang, P. Habisreuther, M. Hansinger, H. Bockhorn, M. Pfitzner, D. Trimis, *Identification of Flame Regimes in Partially Premixed Combustion from a Quasi-DNS Dataset* (Flow, Turbulence and Combustion, 2020)
25. T. Zirwes, T. Häber, F. Zhang, H. Kosaka, A. Dreizler, M. Steinhausen, C. Hasse, A. Stagni, D. Trimis, R. Suntz, H. Bockhorn, *Numerical Study of Quenching Distances for Side-Wall Quenching Using Detailed Diffusion and Chemistry* (Flow, Turbulence and Combustion, 2020)
26. M. Steinhausen, Y. Luo, S. Popp, C. Strassacker, T. Zirwes, H. Kosaka, F. Zentgraf, U. Maas, A. Sadiki, A. Dreizler, C. Hasse, *Numerical Investigation of Local Heat-Release Rates and Thermo-Chemical States in Sidewall Quenching of Laminar Methane and Dimethyl Ether Flames* (Flow, Turbulence and Combustion, 2020)
27. M. Hansinger, T. Zirwes, J. Zips, M. Pfitzner, F. Zhang, P. Habisreuther, H. Bockhorn, The Eulerian stochastic fields method applied to large eddy simulations of a piloted flame with inhomogeneous inlet. Flow, Turbul. Combust. **105**, 837–867 (2020)
28. F. Zhang, T. Zirwes, T. Häber, H. Bockhorn, D. Trimis, R. Suntz, Near wall dynamics of premixed flames, in *Proceedings of the Combustion Institute*, vol. 38 (Elsevier, 2020)
29. Y. Wang, H. Zhang, T. Zirwes, F. Zhang, H. Bockhorn, Z. Chen, Ignition of dimethyl ether/air mixtures by hot particles: impact of low temperature chemical reactions, in *Proceedings of the Combustion Institute*, vol. 38 (Elsevier, 2020)
30. J. Li, Z. Zhao, A. Kazakov, F.L. Dryer, An updated comprehensive kinetic model of hydrogen combustion. Int. J. Chem. Kin. **36**(10), 566–575 (2004)
31. A.C. Hindmarsh, P.N. Brown, K.E. Grant, S.L. Lee, R. Serban, D.E. Shumaker, C.S. Woodward, Sundials: suite of nonlinear and differential/algebraic equation solvers. ACM Trans. Math. Softw. (TOMS) **31**(3), 363–396 (2005)

Computational Fluid Dynamics

Ewald Krämer

The following chapter presents a selection of research projects that were conducted in the field of Computational Fluid Dynamics (CFD) during the reporting period. Numerical simulations were performed on the CRAY XC40 Hazel Hen of the High Performance Computing Center Stuttgart (HLRS) as well as on the ForHLR II of the Steinbuch Centre for Computing (SCC) in Karlsruhe. Preliminary tests could also be carried out on the new HPE Apollo 9000 system "Hawk" at HLRS, which went into operation in 2020 and is expected to exceed the performance of its predecessor by more than a factor of three.

As in previous years, CFD had the strongest demand for supercomputing resources of all disciplines. 72% of the available computing power of Hazel Hen and 45% of ForHLR II were provided to researchers in this field, helping them to expand scientific knowledge in fundamental flow physics or to better understand complex flow behaviour in technical applications. Many important results and new scientific findings have been achieved and remarkable progress has been made. Research has focused on the simulation of fluid dynamic phenomena, which can only be detected by extremely fine temporal and spatial discretization, often in combination with high-order methods, as well as on the optimization of the employed numerical codes with respect to computational performance.

This year, 39 annual reports were submitted and subjected to a peer review process. Research groups from various institutions all over Germany took advantage from the opportunity to utilise the supercomputer resources offered by the two centres. From a large number of excellent reports, the 13 best contributions were selected for publication in this book. The spectrum of projects presented in the following covers fundamental research as well as application-oriented problems of industrial relevance. Challenging fluid dynamic problems are addressed in the fields

E. Krämer
Institute of Aerodynamics and Gas Dynamics, University of Stuttgart,
Pfaffenwaldring 21, 70550 Stuttgart, Germany
e-mail: kraemer@iag.uni-stuttgart.de

of aircraft, rotorcraft and wind turbine aerodynamics, fuel injection, flows over porous and rough surfaces, to name but a few. Different numerical methods, such as Finite Difference, Finite Volume, Smoothed Particle Hydrodynamics, Discontinuous Galerkin, and Lattice Boltzmann, were employed using in-house, commercial or open-source codes.

The first four articles describe the results of so-called Gauss Large Scale Projects, i.e. these projects were granted more than 35 million core-hours per year from the Gauss Centre for Supercomputing (GCS).

Against the backdrop of advancing climate change, a strong focus in aviation research is on the reduction of CO_2 emissions. One means to achieve this goal is to decrease the aircraft's friction drag. A broad variety of passive and active measures has been investigated over the last decades. One technique that has proven to reduce the viscous drag in a turbulent boundary layer is the introduction of spanwise travelling transversal surface waves. In their contribution, *Albers, Meinke, and Schröder* from the Institute of Aerodynamics, RWTH Aachen University, applied this method to two airfoils with different loadings of the boundary layers at a chord-based Reynolds number of 400.000. The airfoil surfaces were actuated in a way that a sinusoidal deflection of the solid walls in wall-normal direction travelling in the spanwise direction was obtained with the actuation parameters chosen time- and chord-dependently. Unsteady highly resolved implicit Large Eddy Simulations were performed to evaluate the impact of the actuation on the aerodynamic efficiency. For both airfoils, viscous as well as pressure drag were reduced significantly, and lift was slightly increased, which resulted in a noticeable improvement of the lift-to-drag ratio. The authors employed their in-house LES code using the MILES approach, which avoids additional explicit modelling of the smallest turbulent scales. The simulations were run on 4,800 cores of Hazel Hen with grid sizes of more than 400 million cells.

A multiphysics solver based on the same approach was applied together with a fully coupled Lagrangian particle tracking model for fuel-spray applications by *Wegmann, Berger, Meinke, and Schröder* from the same institute. Their aim was to study the injection and the mixing behaviour of two biofuels (ethanol and 2-butanone) in an internal combustion engine. The simulations were first validated with experimental Particle Image Velocimetry (PIV) data obtained in a pressure chamber. While the flow and spray formation structures of the two fuels are almost identical in the pressure chamber arrangement, the mixing in the internal combustion engine setup reveals larger differences. The fuel distributions at ignition crank angle are more favourable for a clean combustion for ethanol compared to 2-butanone, because ethanol has a greater axial penetration due to its slower evaporation rate.

Wenzel, Peter, Kloker, and Rist from the Institute of Aerodynamics and Gas Dynamics, University of Stuttgart, and *Vogler* from the HRLS present a performance analysis of the BigWhoop data compression library derived from the JPEF-2000 standard. The compression was applied to DNS data obtained from the simulation of a compressible turbulent boundary layer. The typical size of the output data for the case considered is about one TB per time step on a mesh with 20

billion grid points. Special features of the BigWhoop compressing library are a floating-to-fixed point transform, a discrete wavelet transform to derive a time-frequency representation from the transformed data samples, and an entropy encoding. Different compression rates were tested unveiling a maximum error of 0.2% for a compression of 1:200, which is accurate enough for flow field investigations in the outer boundary layer. For near-wall investigations, the authors recommend to limit the compression rate to 1:100.

The helicopter group at the same institute has long time been successfully engaged in high performance computations of rotorcraft aerodynamics, aeroelastics and aeroacoustics. Since the flow solver FLOWer and the overall toolchain have meanwhile reached a high level of maturity, the main focus has shifted from algorithmic developments to gaining physical understanding of most complex configurations and flight conditions. In their report, *Keßler, Frey, Letzgus, Öhrle, Thiemeier, and Krämer* exemplarily present high-fidelity results for the prediction of pitch-link loads caused by dynamic stall on a full-scale helicopter in turn flight, aeromechanic simulations of compound helicopters taking into account the complex flow interactions as well as the aeroelasticity of the airframe, and a helicopter in manoeuver flight. It is worth noting that the results could partially be validated with flight test data. The detailed analysis of the highly accurate temporal flow fields is a most valuable complement to flight tests (and can replace them in some cases) and helps mitigate the development risks of new configurations. Typically, the employed grids had 100 to 200 million cells and the code was run on up to 7,000 cores. On Hazel Hen, the FLOWer code had demonstrated excellent weak scaling up to 100,000 cores and strong scaling down to less than 10,000 cells per core. Initial tests on the new Hawk system, however, showed that memory bandwidth is a bottleneck due to the higher number of cores per node compared to Hazel Hen, which will likely require substantial reorganisation of data handling.

Steigerwald, Ibach, Reutzsch, and Weigand from the Institute of Aerospace Thermodynamics, University of Stuttgart, used their multiphase DNS solver FS3D to study the impact of a drop onto a film with differing liquid properties with the aim of determining a splashing threshold above which secondary droplet are generated. The discussion of the results in terms of various crown parameters focuses on a detailed, carefully conducted grid study, which shows that a very high grid resolution (1024^3 cells) is required to prevent non-physical generation of secondary droplets. In contrast to many publications on splashing or sloshing, where an "optimal" grid resolution exists (which actually indicates inconsistencies of the models and/or the numerical methods), the results presented by the authors converged towards a consistent solution with increasing mesh refinement. It is particularly noteworthy that the authors present in their report a very comprehensive and detailed analysis of the computing performance showing a significant improvement compared to the last reporting period. This was achieved by optimizing the implemented multigrid solver using an advanced tree structured communication during the solution cycles.

At the same institute, *Chu, Wang, Müller, von Schöning, Liu, and Weigand* study turbulent flows over porous media using DNS. In their contribution, they

focus on a channel whose the lower half is filled with a porous medium. This case is representative for rivers flowing over a porous riverbed, for wind flow over forests, or for technical applications like heat exchangers. Contrary to the standard approach, i.e. the volume-averaged modelling of the porous wall, the microscopic pore structure was fully resolved. The open source high order spectral/hp element solver Nektar++ had been adopted to solve the incompressible Navier-Stokes equations. A low and a high porosity case was considered for two Reynolds numbers and the fluctuations inside the porous wall were analysed by POD. The mesh sizes for these four cases ranged from 250 and 1,100 million cells. The investigations focused on the turbulence modulation and on the energy exchange across the porous surface. The relation between the geometrical characteristics of the porous media and the turbulent structures of the flow field can enable the design of porous structures that generate specific flow properties. Scalability tests on Hazel Hen for the cases under consideration show a maximum efficiency at 10,000 MPI ranks and an efficiency of 70% at 96,000 MPI ranks (the latter corresponding to 5,000 mesh nodes per core).

Yang, Stroh, and Forooghi from the Institute of Fluid Mechanics at the Karlsruhe Institute of Technology performed DNS to study the turbulent flows over a series of irregular rough surfaces in periodic plane channels. It is well known that roughness, in comparison to smooth surfaces, changes the behaviour in the near-wall region resulting in, e.g., a higher skin friction or a higher heat exchange. The generation of the roughness is based on a mathematical algorithm that delivers pseudo-random rough surfaces. The pseudo spectral flow solver SIMSON in conjunction with an immersed boundary method was applied, which is well established for these types of application. One emphasize was on the influence of the channel size, i.e. the reduction of the original size towards the so-called minimal channel, which considerably reduced the computational effort (down to 5–10% CPU-hrs) without deteriorating the quality of the results. The simulations provide a high quality database, e.g. for the development of models that prescribe the effects of roughness on the near-wall flow structures and on the entire flow field. The computations were performed on the ForHLR II.

A simplified approach for the simulation of gusts impacting an aircraft is subject of the article by *Müller, Hillebrandt, Schollenberger, Ehrle, Lutz, and Krämer* from the Institute of Aerodynamics and Gas Dynamics at the University of Stuttgart. The method called Disturbance Velocity Approach superposes the velocities of atmospheric disturbances to the flux balance, thereby neglecting the influence of the aircraft on the gust. Simulations of an airfoil encountering different "1-cos" gusts were performed with simulations in which the gust is fully resolved and propagates with the flow field and which took all interactions into account. Gust wavelength, angle of attack and airfoil shape were varied to identify the respective influences on the accuracy of the method. Additionally, its applicability to gust signals consisting of different superimposed wavelengths was evaluated. The authors show that the simplified approach gives satisfactory results for gust wavelengths larger than the airfoil chord, while significant deviations occur for smaller wavelengths, especially in the pitching moment. Furthermore, the accuracy of the method decreases at high

angles of attack. The simulations were performed on Hazel Hen using the unstructured Finite-Volume Code TAU developed by the German Aerospace Center in unsteady RANS mode. The authors present a very interesting scaling test of the code on the new supercomputer Hawk. Excellent strong scaling is shown up to 256 nodes (32,768 processes), which corresponds to 3,380 grid points per core for the used mesh. Furthermore, they investigated whether the memory bandwidth limits the performance of the code. By using only every second and every forth core on a node, respectively, they found out that there is actually an increase in computational speed when the same number of processes is distributed over more nodes. However, the gain is higher when all cores on a node are used and the number of parallel processes is increased accordingly.

The Wind Energy group of the same institute has been working on a project that studies the effects of wind turbine wakes on the flight dynamics of helicopters during maintenance operations. Under these conditions, the wind turbine is turned out of the wind and rotor and nacelle are encountering wind from the side. This results in a large separation area on leeward side of the nacelle and a big-scale turbulent wake. An approach of the helicopter from this side could cause a significant deterioration of the flight conditions. *Cormier, Letzgus, Lutz and Krämer* present the results of fully-resolved Delayed Detached Eddy Simulations of a stand-still multi-MW offshore wind turbine for different crosswind velocities and analysed the resulting flow fields in the approach and the hovering area. Additionally, the far-wake influence of an operating wind turbine upstream of the target turbine was investigated. While the turbulent kinetic energy and the flow inclination angles do not show particularly high perturbations for the rated wind speed, high variations are observed in the wake of the nacelle for the cut-off wind speed. The predicted flow fields are used in the DLR's real-time flight simulator and in desktop simulations of the project partners at the Technical University of Munich to identify reaction of the helicopter and the pilot. For the time being, the influence of the helicopter on the flow field has not yet been taken into account, but a two-way coupling shall be considered in the future. The simulations were performed on the Hazel Hen with the flow solver FLOWer, using a 5th order WENO scheme in the region of interest. For the code, a good scaling behaviour down to 8,192 cells per core, i.e. 32,000 cores on a mesh with 270 million cells, was demonstrated.

A numerical analysis of the mixing of coolant flows at high temperature differences in a vertical and a horizontal T-junction is presented by *Evrim and Laurien* from the Institute of Nuclear Technology and Energy Systems. Wall-resolved Large-Eddy Simulations were performed using OpenFOAM v5.0, and the results were validated with in-house experimental data showing a very good agreement. Near-wall flow characteristics in terms of the circumferential distribution of the mean normalized temperature as well as of the temperature fluctuations are shown. For the horizontal T-junction, a wavy stable stratification was detected, whereby the upper part of the pipe remains warm during the mixing process. In contrast, the vertical T-junction displays enhanced mixing due a stronger buoyancy effect.

Additionally, the report includes an investigation of the influence of the weld seam on the mixing process. The simulations were run at both the HLRS and the SCC.

Bretl, Trunk, Nirschl, Thäter, and Krause from the Karlsruhe Institute of Technology and *Dorn* from the Federal University of Rio Grande do Sol, Brazil, investigated the settling behaviour of individual randomly shaped particles under extremely high gravity forces. Their aim was to determine the influence of various shape parameters on this behaviour, which can help improve separation processes. For their study, they applied a homogenized lattice Boltzmann method, and it is the first time that a group was able to simulate the sedimentation of a large number (50) of particles with shapes of varying complexity. They used a particle-fluid coupling which is not yet widely used, but appears very promising. Although the particle's mass should be the dominant parameter under the high gravity conditions, they can show that a simple single-parameter regression based on mass cannot successfully describe the observed complex sedimentation and, obviously, shape parameters have to be taken into account. Each simulation ran on 120 cores of ForHLR II at the SCC with a computational domain of 88.5 million cells.

The primary breakup of liquid fuel close behind a spray nozzle typically used in aeroengines was investigated by *Dauch, Chaussonnet, Keller, Okraschevski, Ates, Koch, and Bauer* of the Institute of Thermal Turbomachinery at the Karlsruhe Institute of Technology. They used their in-house weakly compressible Smoothed Particle Hydrodynamics (SPH) method, whose suitability for simulating multiphase flows in technically relevant applications has already been proven by numerous publications. The results presented provide, for the first time worldwide, insights into the film flow on the prefilmer, the associated wetting effects, and the time-dependent formation of ligaments downstream of the atomizing edge. The report focusses on HPC aspects, i.e. the domain decomposition for efficient load balancing and the weak and strong scaling behaviour of the code for the specific application case under consideration. The code is known to have an excellent parallel performance in general, but the authors found that in the present application the performance decreased (but was still very good up to 2,000 cores) due to void spaces inside the bounding box around the fluid domain. This requires a more general domain decomposition method to improve parallel performance. The simulations were run on the ForHLR II of the SCC.

The FLEXI framework for scale resolving fluid dynamics simulations has been developed over the last years by the group of Prof. Munz at the Institute of Aerodynamics and Gas Dynamics, University of Stuttgart. It is based on a high-order Discontinuous Galerkin (DG) method and can treat compressible flow and associated problems like aeroacoustics and particle-flow-interaction. In their article, *Beck, Gao, Kempf, Krais, Kurz, Zeifang, and Munz* from this institute and *Kopper* from the Institute of Aircraft Propulsion Systems report on the progress of the capabilities of the framework towards robust and flexible industrial applications. Three use cases and developments are presented: An implicit time integration scheme for the split DG formulation to solve stiff problems more efficiently (demonstrated for the aeroacoustics of a flow over a cavity), a comparison of LES with DNS data for a compressible turbulent boundary layer, and results of a

1.5-stage turbine simulation with wall-resolved LES using sliding mesh interface. The treatment of the moving mesh causes a decrease in parallel performance of about 20%. Initial tests on the new Hawk system have indicated that the scaling of FLEXI remains stable, but further work is needed to investigate the influence of the reduced memory bandwidth at node level.

The presented selection of projects reflects the continuous progress in High Performance Computing in the field of CFD. Numerical simulations on supercomputers are crucial to gain insight into the often complex, time-dependent flow physics. Some of these flow phenomena occur only on very small scales and can only be brought to light by an extremely fine discretization of the physical domains. Increasingly, unsteady and transient flows are coming to the fore. To make such simulations possible in a reasonable time frame, the most powerful high performance computers are indispensable. But also in industrial design processes, the use of high-fidelity numerical methods on modern supercomputers is becoming more and more important in order to increase the reliability and the efficiency of the simulations, mitigate development time, risks and costs and, thus, increase industrial competitiveness. With the commissioning of the new supercomputer at the HLRS, the computer power available for the users has increased considerably offering them new possibilities. However, initial tests have unveiled that the new architecture requires code adaptations in order to make optimal use of the available resources. A new supercomputer will also be installed at the SCC in the near future. The continuation of the very fruitful collaboration of researchers with experts from the HLRS and the SCC is key to successfully mastering the challenges associated with the transition to the new systems. In this respect, the staffs of both computing centres deserve thanks for their individual and tailor-made support. Without their dedication, it would not have been and will not be possible to maintain the high scientific quality we see in the projects.

Preliminary Study of Particle Settling Behaviour by Shape Parameters via Lattice Boltzmann Simulations

Colin Bretl, Robin Trunk, Hermann Nirschl, Gudrun Thäter, Marcio Dorn, and Mathias J. Krause

Abstract The settling behaviour of 50 generated particle shapes with varying complexity is investigated. The shapes are constructed by randomising parameters of a formula describing solid shapes symmetrical to the axis planes. The simulations are performed with a homogenized lattice Boltzmann method within the framework of the open-source lattice Boltzmann package OpenLB, which proved to be capable of handling arbitrary particle shapes. The setup considers particles in a size range of 100 nm up to 10 µm, for which an efficient separation process can be challenging. Therefore, the case of ultra-centrifugation is considered with 80, 000 times the force of gravity. Although in this setup mass should be the predominant parameter to describe the average settling velocity, an influence of the shape is clearly visible in the results. In particular, the rotational behaviour of the particles is observed showing notable differences in the time it takes for a particle to orient itself towards a stable angle and in the fluctuations around this angle even for similar Reynolds numbers.

Keywords Homogenised lattice-Boltzmann method · Arbitrary shape · OpenLB · Particle sedimentation

C. Bretl · R. Trunk · M. J. Krause (✉)
Lattice Boltzmann Research Group, Karlsruhe Institute of Technology, Straße am Forum 8, 76131 Karlsruhe, Germany
e-mail: mathias.krause@kit.edu

H. Nirschl
Institute for Mechanical Process Engineering and Mechanics, Straße am Forum 8, 76131 Karlsruhe, Germany

G. Thäter
Institute for Applied and Numerical Mathematics, Englerstrasse 2, 76131 Karlsruhe, Germany

M. Dorn
Institute of Informatics, Federal University of Rio Grande do Sul, Av. Bento Goncalves 9500, 91501-970 Porto Alegre, RS, Brazil

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_16

1 Introduction

The influence of a particle's shape on the flow and settling behaviour has been studied by researchers for decades [16, 31]. In the past investigations had to be experimental or were restrained to shapes that are easy to handle. Current studies can take advantage of today's available computing power. New methods are developed and contribute for example to the improvement of processing machines. Fu et al. [4] investigated the influence of particle shape on the behaviour of lactose powder. White and Walton [31] on the other hand studied the influence on the packing density. Especially for particle systems showing a low selectivity in a separation process, simulations of particle dynamics can help to gain knowledge to improve a process. This is for example the case for particles which are too small to be retained by inertial forces and too large to be retained by diffusion [1]. This leads to a size region with a low separation efficiency, usually described as the most penetrating particle size for a filter system. Classification and reduction of this separation gap has been topic of various studies [9, 14, 20]. The aim of this work is the investigation of the influence of various shape parameters on the settling behaviour. This additional information can be utilised to improve the separation processes.

A first challenge is the definition and selection of appropriate shape parameters. Also the feasibility of measuring these parameters for real particle systems is relevant [6]. E.g. Blott and Pye [2] describe the efficiency of a parameter based on the elongation and flatness. However, the use of a single one-dimensional parameter has been shown to be insufficient [15]. Sommerfeld and Qadir [23] considered the sphericity of 8 irregular particles for a Reynolds number between 1 and 200 to derive an improved drag law for non-spherical particles. While the result was in good agreement with experimental data, still effects not described by their model were visible. This implies that a multidimensional approach considering more parameters is required. Therefore in most publications two parameters are considered like sphericity and aspect ratio by van Wachem et al. [28] or the sphericity tangential and normal to the particle's main axis by Hölzer and Sommerfeld [7]. While this work aims at showing the feasibility and necessity for such investigations for arbitrary particles with a size between 100 nm and 10 μm, it is to be seen as a step in the effort to create a model depending on a larger amount of shape parameters and evaluating their impact on the flow behaviour. In addition to the average settling velocity also the orientation of the particle is observed, as the rotation behaviour can change depending on the Reynolds number. This has e.g. been shown by Mandø and Rosendahl [15] for discs and also been investigated by Sheikh et al. [22].

As it is a topic of current research there are various approaches available for simulations of particles in their actual shape without simplifications, except for a discretisation error. Well known is for example the Discrete Element Method (DEM). Its standard form usually considers spheres, however an arbitrary shape can be approximated by gluing several spheres together [3]. Another approach is the immersed boundary method, which depicts the particle by several Lagrangian points on a Eulerian background grid. This however requires interpolations between the particle and

the fluid points [26]. A more comprehensive discussion can be found in [25]. The statement that most publications still only consider particle shapes, which can be easily described by an analytical equation still holds [8, 21, 24, 28, 28], with some exceptions, e.g. Sommerfeld and Qadir [23] considering 8 irregular particle. However, the number of particles is usually low (<20). In addition the gradient smoothed method can be named, which has been applied e.g. by Shao et al. [21], or Grains3D, a package originally meant for the simulation of powders applying a DEM-Distributed Lagrange Multiplier/Fictitious Domain, which has been extended for the simulation of non-convex particle submersed in a fluid [19, 29, 30]. For the lattice Boltzmann method applied here, often a partially saturated cells method [18] is chosen, e.g. applied by Gardner and Sitar [5] for the simulation of settling breakable cubes.

For the simulations the authors apply the Homogenised Lattice Boltzmann Method (HLBM) which has been proposed in [11, 25]. It was chosen as it provides excellent scalability on computing cluster, not affecting the ability to depict arbitrary particle shapes. In comparison to the partially saturated cells methode, it also avoids the challenge of re-initialising cells which change from solid to fluid state since it considers the particles as moving porous media. The simulations are based on the framework of OpenLB [10], an open-source lattice Boltzmann C++ package.

This article is structured as follows. In Sect. 2 the model is described along with the generation of particle shapes. While the discretisation is covered in Sect. 3, Sect. 4 starts with the simulation setup as well as a performance analysis and presentation of the numerical results. In Subsect. 4.4 and 4.5 the results of further statistical investigations of the simulation results are given. A conclusion to the findings is finally given in Sect. 5.

2 Mathematical Modeling

To describe the settling process of a single particle, mainly two components are required. The first one is the fluid it is submersed into, which is described by the Navier–Stokes equation

$$\frac{\partial \boldsymbol{u}_\mathrm{f}}{\partial t} + (\boldsymbol{u}_\mathrm{f} \cdot \nabla)\boldsymbol{u}_\mathrm{f} - \nu \Delta \boldsymbol{u}_\mathrm{f} + \frac{1}{\boldsymbol{\rho}_\mathrm{f}} \nabla p = \boldsymbol{F}_\mathrm{f} \quad \text{in } \Omega \times (t_0, t_1), \tag{1}$$

for a domain Ω and a time interval (t_0, t_1). In this equation $\boldsymbol{u}_\mathrm{f}$ represents the fluid velocity, ν the kinematic viscosity, $\boldsymbol{\rho}_\mathrm{f}$ the fluid's density, p the pressure and $\boldsymbol{F}_\mathrm{f}$ the total force acting on the fluid. The latter is given as the force exerted by the particle in the here considered case.

The particle trajectory as well as rotation is described by Newton's second law of motion

$$m_\mathrm{p} \frac{\partial \boldsymbol{u}_\mathrm{p}}{\partial t} = \boldsymbol{F}_\mathrm{p}, \quad \boldsymbol{J}_\mathrm{p} \frac{\partial \boldsymbol{\omega}_\mathrm{p}}{\partial t} = \boldsymbol{T}_\mathrm{p}. \tag{2}$$

Here m_p is the particle mass, $\boldsymbol{J}_\mathrm{p}$ the moment of inertia, $\boldsymbol{u}_\mathrm{p}$ the particle velocity, $\boldsymbol{\omega}_\mathrm{p}$ the angular velocity and $\boldsymbol{F}_\mathrm{p}$ and $\boldsymbol{T}_\mathrm{p}$ are the force and torque acting on the particle. In the setup given the force consists of an external homogeneous gravitational force and the hydrodynamic forces exerted by the fluid.

2.1 Particle Shape Generation

Usually, particles are approximated by simpler geometries like cubes and spheres, omitting some information which is relevant for the sedimentation process. Since nowadays simulation methods are capable of simulating complex shapes, methods have to be found to describe them. An approach was described by Vogelgesang [27]. Here a slightly modified version of his equation is used, restricting to shapes which are symmetric regarding the principal planes. Thereby, the surface of a 3D particle is described by

$$\left(\sum_{i=1}^{3}\left(\frac{\Delta_i(\mu\theta,\lambda\phi)}{d_i}\right)^p\right)^{\lambda/p} = 1\,, \tag{3}$$

with $\boldsymbol{\Delta}(\theta,\phi)$ being the unit vector according to the spherical coordinates θ and ϕ. The distance vector $\boldsymbol{d}$, the exponent p as well as the parameters for rotational symmetry μ and λ have been randomized within appropriate intervals for particle generation.

3 Lattice Boltzmann Method

In this work the open source lattice Boltzmann framework OpenLB [10], which is developed by the group of the authors, is used. The simulations are performed utilizing the HLBM first published by Krause et al. [11] and developed as well as implemented by the author's group. In this method the particle is depicted as moving porous media and the acting hydrodynamic forces are computed at the surface of the particle via a momentum exchange algorithm, as proposed by Ladd [13]. For further information on the method and implementation details, like the depiction and handling of arbitrary particle shapes, the authors refer to Trunk et al. [25]. Also the implementation has been extensively tested and validated regarding accuracy and large scale simulations, at the author's group [11, 25].

4 Results and Discussion

4.1 Simulation Setup

The previously described adaption of the approach by Vogelgesang [27] was used to generate 150 different particles, of which a subset of 50 is considered in this study.

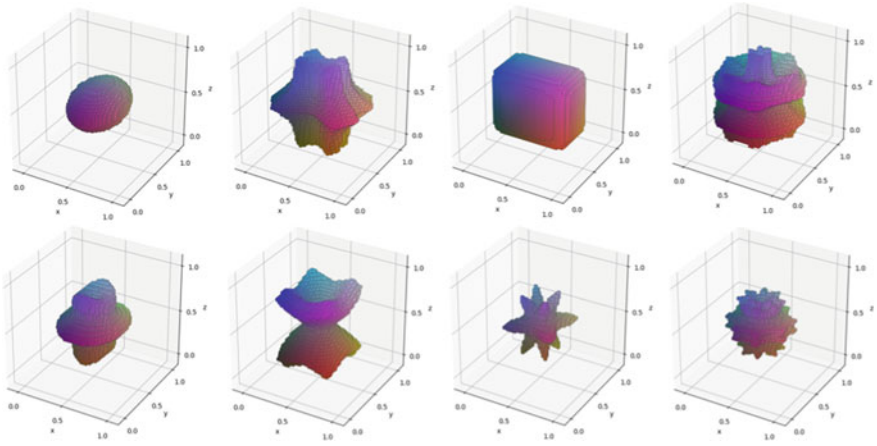


Fig. 1 Some of the particles generated showing both more simple and more complex morphology

He showed that a broad range of polyhedrons can be generated with this approach including convex and star-like particles. The adaptions include restrictions for the occurring parameters $d_i \in (0.15, 0.45)$, $p \in \{2, 4, 8\}$, $\lambda \in \{1, 1.5, 2, 3\}$ and $\mu = 1$ if $\lambda = 1$ and $\mu \in \{1, 1.5, 2, 3\}$. This was done to simplify the generation process and achieve a more homogeneous set of particles regarding the shape. Additionally, the particles were voxelized for simulation purpose. Some exemplary particles of the set can be seen in Fig. 1. The parameters for the whole set can be found in Appendix A.

Particles generated this way are connected and homeomorph to the $\mathbb{R}^3$ standard sphere, i.e. no particle is porous, and have a homogeneous density of 2650 $\frac{\text{kg}}{\text{m}^3}$. During the process of generating particles, care was taken that the particles generated were not too similar in terms of shape. However, this does not guarantee a representative subset of particles. It avoids simulating the settlement of too similar particle though. As the simulation process is computationally expensive, this leads to more insights further evaluating the data.

All sedimentation processes were conducted under the same conditions in a cuboidal simulation area of height $2.8 \cdot 10^{-4}$ m and both width and depth $0.7 \cdot 10^{-4}$ m. The simulated time of each simulation was $3.5 \cdot 10^{-4}$ s. For comparison, each particle fits in a box with edge length $1 \cdot 10^{-5}$ m. Fluid density was set to $998 \frac{\text{kg}}{\text{m}^3}$ and kinematic viscosity to $1.0034 \cdot 10^{-5} \frac{\text{m}^2}{\text{s}}$. For gravitational acceleration, the 80, 000-fold acceleration due to gravity was chosen to shorten computation time. This corresponds to an acceleration in ultra-centrifugation. No other initial forces were applied to the particles. For discretisation the parameters $\delta x = 2.5 \cdot 10^{-7}$ and $\delta t = 3.114 \cdot 10^{-9}$ were used with a lattice relaxation time of $\tau = 0.65$.

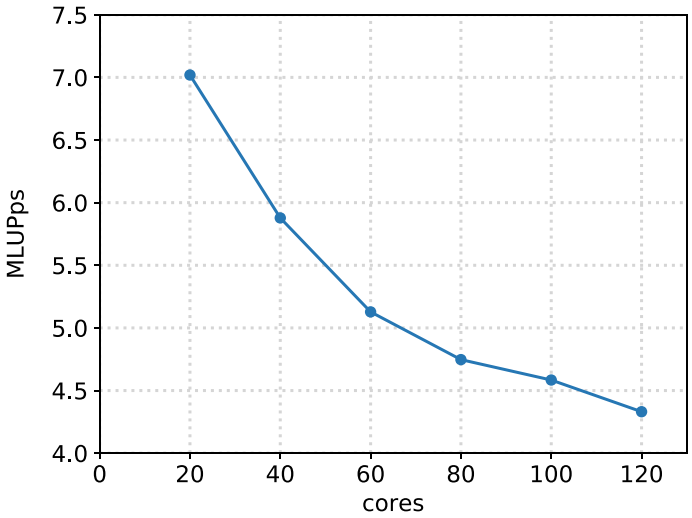


Fig. 2 Parallel scalability of the given setup depicted by million lattice site updates per second and processing core over the number of processing cores

4.2 Performance

The given setup leads to a computational domain of 88, 515, 281 cells with the particle being represented by 4, 664 cells. The performance is measured in million lattice site updates per second and processing core (MLUPps). Utilizing MPI-parallelism a single simulation takes about 5.36 h on 120 CPU-cores on a system equipped with Intel Xeon Silver 4114 CPUs. The scaling given in Fig. 2 shows a behaviour similar to previous performance tests of OpenLB considering the case of a lid driven cavity [12].

For the calculation of the speedup $S_\mathrm{p} = t(1)/t(p)$ on p processors, the time of a run on 20 cores was multiplied by 20 and used as reference time $t(1)$, as the duration for a run on a single core would exceed a week. In Fig. 3 a comparison between the speedup of the HLBM implemented by the author's group and a theoretical code, of which 99.5% can be run in parallel, is shown. For the latter the speedup is computed according to

$$S_\mathrm{p} = \frac{t(1)}{t(1) \cdot ((1 - 0.995) + \frac{0.995}{p})}. \tag{4}$$

It will remain challenging to improve the speed of a single simulation. However, larger HPC systems may allow the simulation of particle swarms instead of single particles in the future [17].

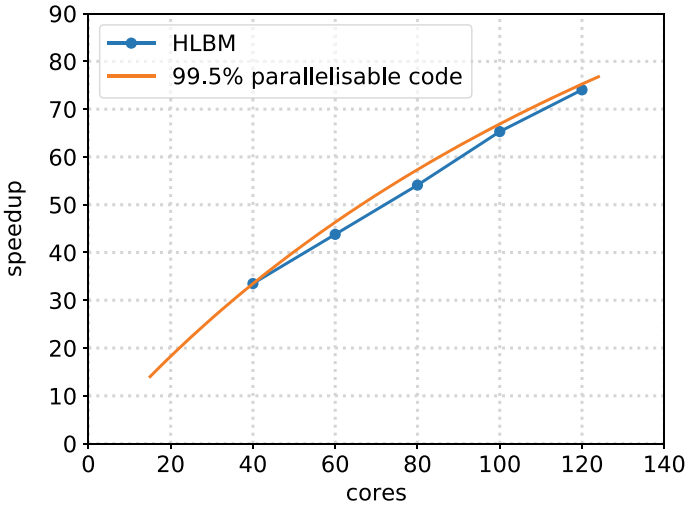


Fig. 3 The speedup observed in the code implemented by the authors is compared to a theoretical code which is 99.5% parallelisable

4.3 Numerical Results

The results of the simulations of settling particles suggest that a particle undergoes several phases during sedimentation as one would expect. During a phase of acceleration the particle starts tumbling, depending on the shape. Not all particles reach a phase of stable fall afterwards, which then lasts until impact, see Fig. 4. For example, the vector of orientation $\boldsymbol{\theta}$ of particle 41 stabilises pretty fast and it sediments almost steadily in terms of $|\boldsymbol{\omega}|$ where $\boldsymbol{\omega}$ depicts the angular velocity of the particle. Particle 36 on the other hand seems to trundle and does not become stable in terms of $|\boldsymbol{\omega}|$. The decrease of $|\boldsymbol{\omega}|$ also is less smooth compared to the one of particle 41. Comparison of those two particle shows the importance of an investigation of the early stabilisation phase during sedimentation, as frequent collisions in suspensions deflect the particle from their equilibrium state.

A particle was declared to be stable, if the difference of $|\boldsymbol{\omega}|$ in two consecutive time steps, falls below $0.1 \cdot |\boldsymbol{\omega}|_{\max}$ with $|\boldsymbol{\omega}|_{\max}$ being the maximal angular velocity norm before impact. The point of impact was declared to be the point, when a particle was less than $1 \cdot 10^{-5}$ m above the ground to mask out wall influences.

4.4 Statistical Investigation

A linear regression was performed to investigate whether the terminal velocity of the particle can be predicted only by their volume. Without considering buoyancy effects the terminal velocity of a particle

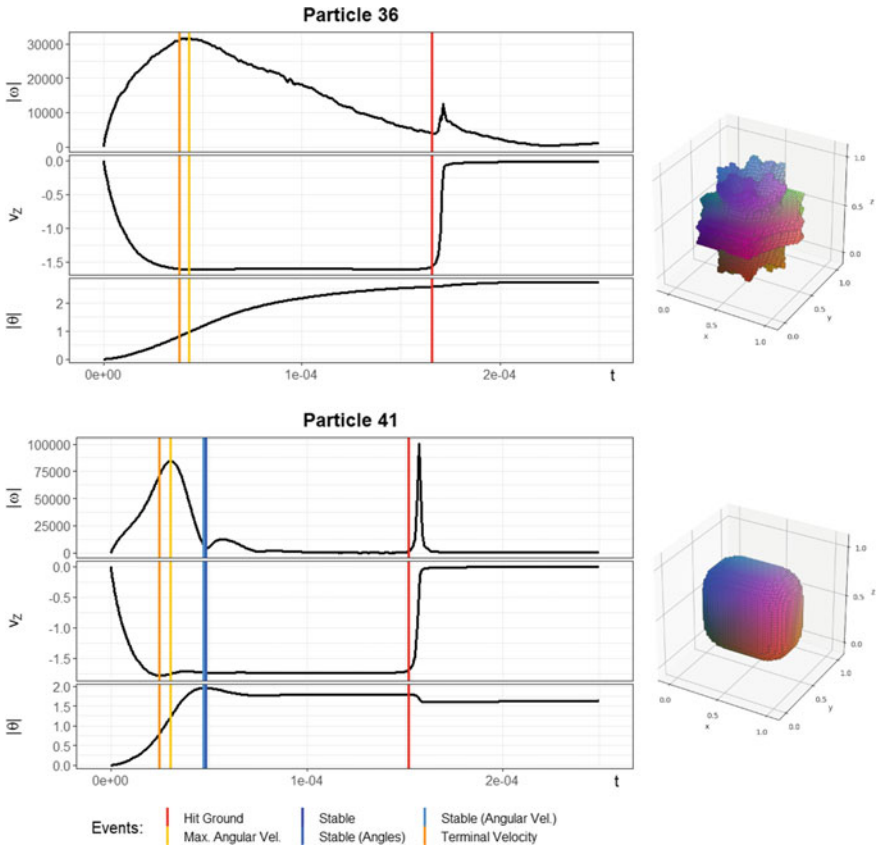


Fig. 4 Aspects of sedimentation during the simulated fall of two particles. Stability was determined both regarding the angles and the angular velocities. The dark blue line depicted with "Stable" is the maximum of both expressing a conservative understanding of stability

$$v_\mathrm{t} = \sqrt{\frac{2m_\mathrm{p}g}{\rho A C_\mathrm{d}}}, \tag{5}$$

can be expressed via the mass of the particle m_p, the gravitational acceleration g, the drag coefficient C_d, the density of the fluid it is falling through $\boldsymbol{\rho}_\mathrm{f}$ and the projected area A of the particle.

Geometrical influences put aside, the terminal velocity should mainly be determined by the square rooted mass of a particle. The reason being that sedimentation of nano-particles was simulated at 80, 000 g, thus the numerator in Eq. 5 dominates v_t.

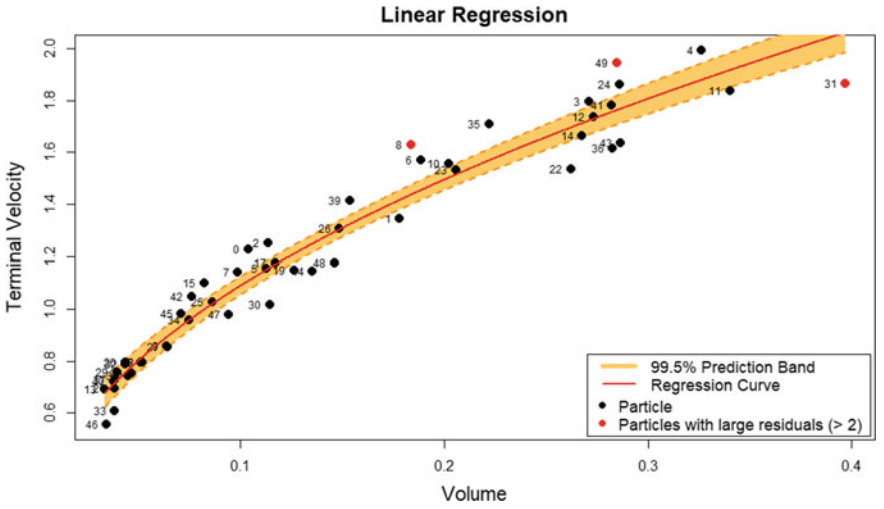


Fig. 5 Linear regression predicting terminal velocity by square rooted mass. Although mass should be dominant, the model considers 3 particles (6%) as outliers

A linear regression with $\sqrt{m_\mathrm{p}}$ as single regressor is therefore able to detect volume as a highly significant regressor with a p-value below $2 \cdot 10^{-16}$. Figure 5 pictures the regression curve of said model where the maximal velocity before impact was used as terminal velocity. Shown along with the regression curve is the 99.5% prediction band of the model.

According to the one in ten rule, the 50 observations should be sufficient to reliably fit a linear model with $\sqrt{m_\mathrm{p}}$ as single regressor if they are a representative subset of particles.

A Kolmogorov-Smirnov test however rejects the hypothesis of the particles being uniformly distributed regarding volume on a 5% level. Besides them being generated randomly, there are more small particles. Yet, even more conservative rules like a one in 20 or one in 50 rule would not suggest that there is too little data for fitting a single regressor. We therefor consider the set of 50 particle to be adequate for fitting a least square model.

Statistical tests whether the assumptions of a linear model are fulfilled were performed. A Shapiro-Wilk test regarding the distribution of the standardised residuals would not reject the hypothesis of them not being normally distributed with a p-value of 0.80. Furthermore, a Durbin-Watson-Test with an outcome of 1.96 does not detect any first order autocorrelation among the residuals. What appears to be an issue though is the heteroscedasticity notable in Fig. 5: the scattering of larger particles seems to be higher than the scattering of smaller particles.

A usual approach in such a situation includes transforming the data by taking their logarithms or square roots. From Eq. (5) we have a physically motivated relationship at hand, thus transforming the data appears to be inappropriate. From a statistical point of view, this has the following consequence: the least square model used is not the best linear unbiased estimator for the regression coefficients any longer. This can

help to understand why the terminal velocity of several particles differs notably from the one the model would predict.

There ought to be other aspects determining v_t even if it is investigated in a setting as ours, where volume should be the dominant factor. From Eq. (5) we expect them to be associated with the overall shape of a particle. Thus, an outlier analysis was performed.

4.5 Outlier Analysis

Regarding the linear regression approach in Subsect. 4.4 particle 8, 31 and 49 appear to be outliers based on the absolute values of their corresponding standard residuals being larger than 2. Among them, particle 31 has the largest residual, it is depicted in Fig. 6 along with a snapshot of the velocity field. When referring to them as outliers, we implicitly refer to them as statistical outliers. All data involved in the regression are the result of numerical simulations and computer-aided analysis, so there is no reason to expect measuring errors or the like for single particles.

It should be mentioned that it is common to consider observations with standard residuals larger than 3 as outliers. However, theoretical insights like Eq. (5) give us

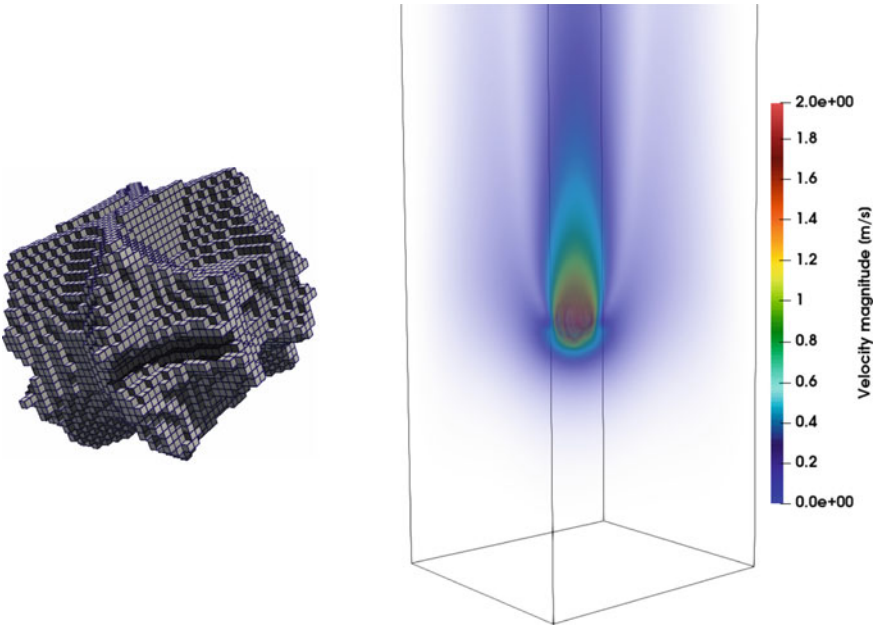


Fig. 6 Geometric representation of particle 31 within the simulation (left) and velocity field around the settling particle at $t \approx 10^{-4}$ s (right)

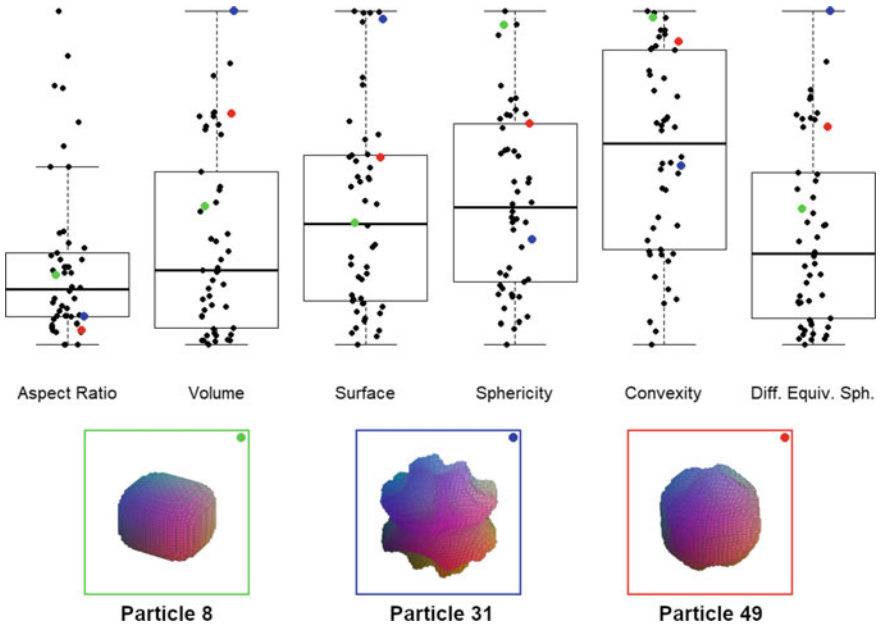


Fig. 7 Box plots of several shape parameters. The outlying particles are highlighted in green (particle 8), blue (particle 31) and red (particle 49)

a clear idea of what the model should look like. Moreover, we have an appropriate number of observations for fitting a single regressor as briefly discussed in Subsect. 4.4. Therefore we work with the stricter characterisation of outliers.

Figure 7 shows the shape of particle 8, 31 and 49. How several aspects of shape are distributed is displayed via box plots. The outlying particle 8, 31 and 49 are highlighted. It also shows a comparison of the terminal velocity of the particle compared to the one expected from a sphere with the same volume as the particle.

One has to be careful with particle 31, because it is the largest one generated. It is expected to be a leveraging observation, especially since the particles are not uniformly distributed regarding volume and classical linear models being sensitive to outliers. One also should have in mind that classical linear regression is known to be prone to overfitting. With only a single regressor and 50 observations considered, overfitting should be less of an issue in our case.

Particle 49 on the other side belongs to a group of nine similar-sized particles. When discussing it as an outlier, the fact that the region of similar-sized particles is sub-scanned (as for particle 31) is less relevant.

Particle 8 stands out as being one of the two most spherical and convex particles. Compared to other particles, there is no big difference between the terminal velocity of a sphere with the same mass as particle 8 and the terminal velocity observed (within both quartiles). This seems to be reasonable since particle 8 is closer to a sphere as the large value for sphericity indicates.

Particle 31 is the heaviest of the 50 particle investigated. It is slightly less convex and spherical but does not stand out in terms of aspect ratio, sphericity and convexity. However, it looks way more irregular than for example particle 8. The difference between its terminal velocity and the one an equivalent sphere of particle 31 would have is the largest among all particle. Additionally, the point it becomes stable expressed via the angular velocity lies pretty close to its impact point.

Particle 49 is more spherical and convex than 75% of the particles and it also has a lower aspect ratio than 75% of the particle.

5 Conclusion

In this article, the simulated settling behaviour of 50 particles was studied with 80, 000 times the force of gravity. Although in this setup mass should be the predominant parameter to describe settling velocity v_t, influence of the overall shape is still visible in the results. A linear regression model with $\sqrt{v_t}$ as single regressor should be able to explain the settling behaviour due to what is physically known about sedimentation, however, three of the 50 particle have shown up as statistical outliers.

The results show the need for further studies. On one hand the data pool has to be improved by consideration of a larger set of particles, whose shape parameters are distributed more homogeneously. On the other hand appropriate shape parameters have to be selected and considered as regressors. It has to be investigated which aspects of shape or morphology cause the deviation. Moreover, there are other aspects of sedimentation like spinning and stabilisation during fall which are influenced by aspects of shape. For example, our results indicate that particle 36 and 41 show some notable differences in their sedimentation behaviour though they are very similar in terms of mass.

Acknowledgements The authors would also like to thank the Steinbuch Centre for Computing at KIT for providing access to their high performance computer ForHLR II, where most computations were carried out as part of the CPE project.
The research leading to this results was supported by the German Research Foundation (DFG) [grant KR 4259/8-1] within the priority program SPP2045 MehrDimPart “Highly specific and multidimensional fractionation of fine particle systems with technical relevance”.

Conflict of Interest The authors declare that they have no conflict of interest.

A Appendix: Parameter for Particle Shape Generation

ID	p	λ	μ	d_x	d_y	d_z
0	2	1	1	0.391	0.279	0.230
1	8	1.5	1.5	0.157	0.346	0.435
2	2	1	1	0.356	0.242	0.313
3	8	1	1	0.261	0.363	0.367
4	4	1.5	2	0.385	0.448	0.261
5	2	1	1	0.434	0.205	0.304
6	2	3	1.5	0.403	0.285	0.389
7	4	2	2	0.207	0.329	0.241
8	4	1	1	0.301	0.350	0.267
9	4	3	1.5	0.252	0.304	0.155
10	4	1.5	1.5	0.231	0.341	0.393
11	8	1.5	2	0.258	0.441	0.410
12	8	1	1	0.233	0.414	0.365
13	2	1	1	0.245	0.156	0.211
14	8	3	1.5	0.298	0.233	0.444
15	2	1.5	2	0.237	0.302	0.278
16	2	3	1.5	0.321	0.158	0.255
17	2	3	1	0.422	0.332	0.238
18	4	2	1	0.307	0.160	0.181
19	8	2	2	0.188	0.421	0.238
20	4	1.5	1.5	0.167	0.190	0.214
21	2	1.5	1.5	0.301	0.157	0.228
22	8	3	1	0.422	0.421	0.247
23	4	1.5	1	0.342	0.245	0.428
24	4	1.5	1	0.437	0.433	0.205
25	2	2	1	0.267	0.417	0.215
26	8	2	1.5	0.316	0.330	0.223
27	4	3	1.5	0.324	0.239	0.169
28	4	2	2	0.286	0.152	0.183
29	2	1	1	0.163	0.185	0.311
30	4	3	1.5	0.450	0.328	0.171
31	8	2	1.5	0.275	0.412	0.422
32	4	3	1	0.196	0.210	0.181
33	2	3	2	0.174	0.376	0.167
34	8	3	2	0.258	0.232	0.195
35	2	3	1	0.398	0.300	0.409
36	8	3	1.5	0.446	0.367	0.291
37	2	1.5	1.5	0.150	0.252	0.216
38	8	1.5	1	0.153	0.168	0.278
39	2	2	2	0.306	0.222	0.426
40	8	2	2	0.161	0.152	0.199
41	4	1	1	0.251	0.431	0.406
42	2	1	1	0.243	0.185	0.400
43	8	3	1.5	0.427	0.381	0.290
44	4	3	1.5	0.391	0.390	0.181
45	2	1	1	0.298	0.161	0.354
46	2	3	1	0.162	0.397	0.154
47	2	3	1.5	0.171	0.179	0.427
48	8	2	1	0.428	0.394	0.151
49	2	3	2	0.436	0.358	0.423

References

1. R. Baker, *Membrane Technology and Applications* (Wiley, Hoboken, 2004). https://doi.org/10.1002/0470020393
2. S. Blott, K. Pye, Particle shape: a review and new methods of characterization and classification. Sedimentology **55**(1), 31–63 (2008). https://doi.org/10.1111/j.1365-3091.2007.00892.x
3. J. Favier, M. Abbaspour-Fard, M. Kremmer, A. Raji, Shape representation of AXI-symmetrical, non-spherical particles in discrete element simulation using multi-element model particles. Eng. Comput.: Int. J. Comput.-Aided Eng. **16**(4), 467–480 (1999). https://doi.org/10.1108/02644409910271894
4. X. Fu, D. Huck, L. Makein, B. Armstrong, U. Willen, T. Freeman, Effect of particle shape and size on flow properties of lactose powders. Particuology **10**(2), 203–208 (2012). https://doi.org/10.1016/j.partic.2011.11.003
5. M. Gardner, N. Sitar, Coupled three-dimensional discrete element-lattice Boltzmann methods for fluid-solid interaction with polyhedral particles. Int. J. Numer. Anal. Meth. Geomech. **43**(14), 2270–2287 (2019). https://doi.org/10.1002/nag.2972
6. M. Hentschel, N. Page, Selection of descriptors for particle shape characterization. Part. Part. Syst. Charact. **20**(1), 25–38 (2003). https://doi.org/10.1002/ppsc.200390002
7. A. Hölzer, M. Sommerfeld, New simple correlation formula for the drag coefficient of non-spherical particles. Powder Technol. **184**(3), 361–365 (2008). https://doi.org/10.1016/j.powtec.2007.08.021
8. A. Hölzer, M. Sommerfeld, Lattice Boltzmann simulations to determine drag, lift and torque acting on non-spherical particles. Comput. Fluids **38**(3), 572–589 (2009). https://doi.org/10.1016/j.compfluid.2008.06.001
9. C. Jung, H.S. Park, Y. Kim, Theoretical study for the most penetrating particle size of dust-loaded fiber filters. Sep. Purif. Technol. **116**, 248–252 (2013). https://doi.org/10.1016/j.seppur.2013.05.045
10. M. Krause, S. Avis, D. Dapalo, N. Hafen, M. Haußmann, M. Gaedtke, F. Klemens, A. Kummerländer, M.L. Maier, A. Mink, J. Ross-Jones, S. Simonis, R. Trunk, OpenLB release 1.3: open source lattice boltzmann code (2019). https://doi.org/10.5281/zenodo.3625967. https://doi.org/10.5281/zenodo.3625967
11. M. Krause, F. Klemens, T. Henn, R. Trunk, H. Nirschl, Particle flow simulations with homogenised lattice Boltzmann methods. Particuology **34**, 1–13 (2017). https://doi.org/10.1016/j.partic.2016.11.001
12. M.J. Krause, A. Kummerländer, S.J. Avis, H. Kusumaatmaja, D. Dapelo, F. Klemens, M. Gaedtke, N. Hafen, A. Mink, R. Trunk, J.E. Marquardt, M.L. Maier, M. Haussmann, S. Simonis, OpenLB–open source lattice Boltzmann code. Comput. Math. Appl. (2020). https://doi.org/10.1016/j.camwa.2020.04.033
13. A.J.C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation. J. Fluid Mech. **271**, 285–309 (1994). https://doi.org/10.1017/s0022112094001771
14. K. Lee, B. Liu, On the minimum efficiency and the most penetrating particle size for fibrous filters. J. Air Pollut. Control Assoc. **30**(4), 377–381 (1980). https://doi.org/10.1080/00022470.1980.10464592
15. M. Mandø, L. Rosendahl, On the motion of non-spherical particles at high reynolds number. Powder Technol. **202**(1–3), 1–13 (2010). https://doi.org/10.1016/j.powtec.2010.05.001
16. J. McNown, J. Malaika, Effects of particle shape on settling velocity at low reynolds numbers. Trans. Am. Geophys. Union **31**(1), 74 (1950). https://doi.org/10.1029/tr031i001p00074
17. M. Mohrhard, G. Thäter, J. Bludau, B. Horvat, M. Krause, Auto-vectorization friendly parallel lattice Boltzmann streaming scheme for direct addressing. Comput. Fluids **181**, 1–7 (2019). https://doi.org/10.1016/j.compfluid.2019.01.001
18. D. Noble, J. Torczynski, A lattice-Boltzmann method for partially saturated computational cells. Int. J. Mod. Phys. C **9**(8), 1189–1201 (1998). https://doi.org/10.1142/s0129183198001084

19. A. Rakotonirina, J.Y. Delenne, F. Radjai, A. Wachs, Grains3d, a flexible dem approach for particles of arbitrary convex shape–part III: extension to non-convex particles modelled as glued convex particles. Comput. Part. Mech. **6**(1), 55–84 (2019). https://doi.org/10.1007/s40571-018-0198-3
20. E. Schach, M. Buchmann, R. Tolosana-Delgado, T. Leißner, M. Kern, K. van den Boogaart, M. Rudolph, U. Peuker, Multidimensional characterization of separation processes—part 1: introducing kernel methods and entropy in the context of mineral processing using SEM-based image analysis. Miner. Eng. **137**, 78–86 (2019). https://doi.org/10.1016/j.mineng.2019.03.026
21. B. Shao, G. Liu, T. Lin, G. Xu, X. Yan, Rotation and orientation of irregular particles in viscous fluids using the gradient smoothed method (GSM). Eng. Appl. Comput. Fluid Mech. **11**(1), 557–575 (2017). https://doi.org/10.1080/19942060.2017.1329169
22. M. Sheikh, K. Gustavsson, D. Lopez, E. Lévêque, B. Mehlig, A. Pumir, A. Naso, Importance of fluid inertia for the orientation of spheroids settling in turbulent flow. J. Fluid Mech. **886** (2020). https://doi.org/10.1017/jfm.2019.1041
23. M. Sommerfeld, Z. Qadir, Fluid dynamic forces acting on irregular shaped particles: simulations by the lattice-Boltzmann method. Int. J. Multiph. Flow **101**, 212–222 (2018). https://doi.org/10.1016/j.ijmultiphaseflow.2018.01.016
24. S. Tao, Q. He, B. Chen, X. Yang, S. Huang, One-point second-order curved boundary condition for lattice Boltzmann simulation of suspended particles. Comput. Math. Appl. **76**(7), 1593–1607 (2018). https://doi.org/10.1016/j.camwa.2018.07.013
25. R. Trunk, J. Marquardt, G. Thäter, H. Nirschl, M. Krause, Towards the simulation of arbitrarily shaped 3d particles using a homogenised lattice Boltzmann method. Comput. Fluids **172**, 621–631 (2018). https://doi.org/10.1016/j.compfluid.2018.02.027
26. M. Uhlmann, An immersed boundary method with direct forcing for the simulation of particulate flows. J. Comput. Phys. **209**(2), 448–476 (2005). https://doi.org/10.1016/j.jcp.2005.03.017
27. R. Vogelgesang, Global surface parameterization by smooth facet selection. J. Comput. Theor. Nanosci. **8**(8), 1631–1638 (2011). https://doi.org/10.1166/jctn.2011.1858
28. B. van Wachem, M. Zastawny, F. Zhao, G. Mallouppas, Modelling of gas-solid turbulent channel flow with non-spherical particles with large stokes numbers. Int. J. Multiph. Flow **68**, 80–92 (2015). https://doi.org/10.1016/j.ijmultiphaseflow.2014.10.006
29. A. Wachs, A DEM-DLM/FD method for direct numerical simulation of particulate flows: sedimentation of polygonal isometric particles in a newtonian fluid with collisions. Comput. Fluids **38**(8), 1608–1628 (2009). https://doi.org/10.1016/j.compfluid.2009.01.005
30. A. Wachs, L. Girolami, G. Vinay, G. Ferrer, Grains3D, a flexible dem approach for particles of arbitrary convex shape—part I: numerical model and validations. Powder Technol. **224**, 374–389 (2012). https://doi.org/10.1016/j.powtec.2012.03.023
31. H. White, S. Walton, Particle packing and particle shape. J. Am. Ceram. Soc. **20**(1–12), 155–166 (1937). https://doi.org/10.1111/j.1151-2916.1937.tb19882.x

Towards the Numerical Determination of the Splashing Threshold of Two-Component Drop Film Interactions

Jonas Steigerwald, Matthias Ibach, Jonathan Reutzsch, and Bernhard Weigand

Abstract The scenario of an impacting drop onto a film is highly relevant in many natural and technical systems. A fundamental and often required parameter of these interactions is the so called splashing threshold above which secondary droplets are generated. For interactions with differing liquid properties for the film and the impacting drop a general splashing threshold is, however, still unknown because an experimental determination is difficult to achieve. For this reason, we investigate the suitability of a numerical determination of this threshold by means of direct numerical simulation using the multiphase flow solver Free Surface 3D (FS3D). Simulations across an already existing splashing threshold are performed stemming from an empirical correlation. In order to determine the necessary grid resolution for accurately reproducing the corresponding impact regime, all interactions are simulated for several grids. A detailed grid study shows that only by using very high grid resolutions the threshold can be reproduced with a sufficient accuracy, whereas the use of coarser resolutions leads to a significant underestimation of the threshold. Additionally, simulations of highly resolved impact phenomena on thin films depend heavily on the efficient solution of the problem with most of the computational costs affiliated to solving the Pressure Poisson Equation within the FS3D framework. Therefore, the implemented multigrid solver was optimized employing advanced tree structured communication during coarsening and refinement on the levels during the solution cycle. A performance analysis of FS3D using the original and the improved multigrid solver shows that the implemented tree structured communication leads to a remarkable speed-up.

1 Introduction

The interaction of a drop with a wetted surface is a fundamental process in many natural systems and technical applications. The outcome of such interactions is often

J. Steigerwald (✉) · M. Ibach · J. Reutzsch · B. Weigand
Institute of Aerospace Thermodynamics (ITLR), University of Stuttgart, Pfaffenwaldring 31, 70569 Stuttgart, Germany
e-mail: jonas.steigerwald@itlr.uni-stuttgart.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_17

decisive like for the quality of ink-jet printers or for the environmental impact in automotive engines. In diesel engines with implemented post-injection technology, for example, fuel drops impact onto the cylinder wall wetted with lubrication oil. Its removing from the wall due to impacting fuel drops increases the pollutant emissions, since oil is becoming part of the combustion process. Therefore, a precise knowledge of the threshold impact energy at which new secondary droplets are generated, which is called splashing, is of crucial importance for both one- and two-component drop film interactions.

A schematic setup of a drop film interaction with its fundamental process parameters is depicted in Fig. 1 on the left-hand side, where D denotes the drop diameter, U the impact velocity, h the film thickness and g the gravitational acceleration. Further relevant physical properties are the density ρ, the dynamic viscosity μ and the surface tension σ, where the subscripts D, F and g denote the liquid of the drop, the liquid of the film and the surrounding gaseous phase. Figure 1 also illustrates the impact outcomes deposition (middle; no generation of secondary droplets) and splashing (right). Geometrical crown characteristics like height H_{CR}, base diameter D_{Base} and angle α are shown as well. The regime in between deposition and splashing, which is not shown here, is called transition zone.

For one-component drop film interactions with a density ratio $\chi = \rho_D/\rho_F = 1$, a viscosity ratio $\lambda = \mu_D/\mu_F = 1$ and a surface tension ratio $\gamma = \sigma_D/\sigma_F = 1$, there exists a large amount of empirical splashing threshold correlations. An overview can be found e.g. in [10, 15]. An often used definition first proposed by Mundo et al. [17] reads

$$K = Oh^{-1/4}\, We^{5/8}, \tag{1}$$

where $Oh = \mu/\sqrt{\rho\sigma D}$ denotes the Ohnesorge number and $We = \left(\rho D U^2\right)/\sigma$ the Weber number. The same functional relationship was also derived by Gomaa [11] by means of dimensional analysis. Since the threshold is also influenced by the film thickness h, the critical value K_{crit} is typically defined with an additional dependence $K_{crit} = K_{crit}(\delta)$, where $\delta = h/D$ denotes the non-dimensional film thickness. Therefore, splashing takes place if $K > K_{crit}$. Besides experimental investigations, numerical studies were performed as well by Maichle [16] and Gomaa [11] in order

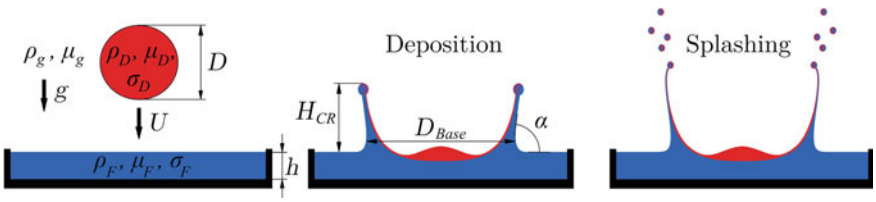


Fig. 1 Schematic setup of drop film interaction (left) and schematic sketch of the impact outcomes deposition (middle) and splashing (right)

to determine the threshold for specific parameter spaces. Both employed the CFD multiphase flow solver Free Surface 3D (FS3D), which has already been used successfully for simulating drop film interactions for more than 20 years (Rieber and Frohn [24]). Thus, this in-house code of ITLR is also used in the presented study.

In contrast to one-component interactions, there have been very few studies for two-component interactions, where the liquids of drop and film differ. A first empirical threshold correlation of a two-component system was proposed by Geppert et al. [9]. In their correlation

$$Oh_{\overline{\mu}}^{-1/4} We_D^{5/8} = 114 + 163\delta^{6/5} \tag{2}$$

they used for Oh and We the properties of the drop liquid with the exception of the dynamic viscosity, which is chosen as arithmetic mean $\overline{\mu} = 0.5\,(\mu_D + \mu_F)$. Since they used only two liquids in their experimental campaign, the correlation is only valid for binary systems with a specific χ and γ. The correlation

$$\overline{Oh}\,\overline{Re}^{1.29} = \left(4400 + 8900\delta^{1.44}\right)^{0.625}, \tag{3}$$

where $Re = \sqrt{We}/Oh$ denotes the Reynolds number, was proposed by Bernard et al. [2], who tried to derive from own experimental data and already existing correlations a single threshold correlation for both one- and two-component drop film interactions. They defined Oh and Re with averaged physical properties.

The missing property ratios χ, λ and γ in both correlations mentioned above demonstrate the impossibility of an experimental determination of the threshold for general two-component drop film interactions, simply due to the limited amount of available liquids and their fixed physical properties. A determination of such a threshold can, therefore, only be achieved by means of a numerical parameter study.

The first study about numerical determination of the splashing threshold using two different liquids was performed by Copplestone [3], who used FS3D with a simplified multi-component extension. However, the results showed discrepancies of up to 50%. In this paper, we want to clarify the question of suitability of FS3D to determine the splashing threshold of two-component drop film interactions by using an improved multi-component extension by Kaufmann et al. [13]. Therefore, simulations with impact outcomes across the above mentioned splashing thresholds, equations (2) and (3), are performed. For all simulations the numerically challenging case of a density ratio $\chi > 1$ is used, something that, to the best of our knowledge, has not been investigated in the literature before. A grid study of the case close to the splashing threshold is presented, which shall help to identify the limitations and the numerical difficulties when reducing the resolution of the computational grid. Furthermore, an extensive numerical study requires a good overall performance of the solver used. For this reason, we optimized the parallel performance of the central element of FS3D, the multigrid (MG) solver being responsible for most of the computational

time. By taking advantage of a tree structured communication pattern inspired by the work of Ertl et al. [6], the parallel performance can be increased dramatically, which is shown in a detailed performance analysis.

2 Fundamental Method and Computational Setup

FS3D is a CFD code for direct numerical simulation (DNS) of incompressible multiphase flows, which was developed at the Institute of Aerospace Thermodynamics (ITLR) in Stuttgart and has been continuously enhanced in the last 20 years. A wide variety of recent studies to drop impacts [1, 13], jet atomization [7] as well as to phase change processes [18, 19, 21] shows the ability of FS3D to simulate highly dynamic processes resulting in complex topologies.

2.1 Mathematical Description and Numerical Approach

FS3D solves the incompressible Navier-Stokes equations together with the equation for mass conservation

$$\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) = 0, \tag{4}$$

$$\partial_t (\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = \nabla \cdot (\mathbf{S} - \mathbf{I}p) + \rho \mathbf{g} + \mathbf{f}_\gamma \tag{5}$$

on finite volumes (FV), where $\mathbf{u}$ denotes the velocity vector, p the static pressure, $\mathbf{g}$ the acceleration due to gravity, $\mathbf{S}$ the shear stress tensor and $\mathbf{I}$ the identity matrix. The term $\mathbf{f}_\gamma$ accounts for body forces which is used to model surface tension at the interface in multiphase flows. The flow field is computed by solving Eqs. (4) and (5) in a one-field formulation, where the different phases consisting of possibly different compositions are regarded as a single fluid with variable physical properties. In order to identify different phases, the volume-of-fluid (VOF) method by Hirt and Nichols [12] is used, in which an additional variable f is introduced. The variable f determines the liquid volume fraction in each control volume and is defined as

$$f(\mathbf{x}, t) = \begin{cases} 0 & \text{outside the liquid phase,} \\ (0, 1) & \text{at the interface,} \\ 1 & \text{in the liquid phase.} \end{cases} \tag{6}$$

The scalar f is advected across the computational domain by using the transport equation

$$\partial_t f + \nabla \cdot (f \mathbf{u}) = 0. \tag{7}$$

To keep the interface sharp, the corresponding f fluxes over the sides of each control volume are calculated by using the piecewise linear interface calculation (PLIC)

method by Rider and Kothe [22]. For the distinction between different liquids, additional VOF variables $\psi_i = V_i / V$ are introduced, representing the volume fraction of species i inside the liquid phase. The scalars ψ_i are advected analogously to f with

$$\partial_t f\psi_i + \nabla \cdot (f\psi_i \mathbf{u}) = 0, \tag{8}$$

whereby the liquids can interpenetrate each other. In such a case, a linear mixing behavior without volume change is assumed. Due to the high impact velocity, diffusion between the species can be neglected explaining the zero right-hand side of Eq. (8). Furthermore, the surface tension forces are modeled by the conservative continuous surface stress (CSS) model by Lafaurie et al. [14].

Consistent with the one-field formulation, the local physical properties are defined by using the volume fraction f, the species volume fractions ψ_i and the respective values of the ambient gas and the liquids of drop and film. In the present case, the local density is calculated with

$$\rho(\mathbf{x}, t) = \rho_l f(\mathbf{x}, t) + \rho_g (1 - f(\mathbf{x}, t)) \qquad \text{with} \qquad \rho_l = \rho_D \psi_D + \rho_F \psi_F, \tag{9}$$

where the index l denotes liquid and g denotes the gas phase. The calculation of the liquid mixture density ρ_l as weighted arithmetic mean ensures mass conservation due to the zero excess volume of the mixture. The local viscosity

$$\mu(\mathbf{x}, t) = \mu_l f(\mathbf{x}, t) + \mu_g (1 - f(\mathbf{x}, t)) \tag{10}$$

is calculated by using the volume fraction based viscosity mixture model for binary liquid mixtures by Dey and Biswas [4]

$$\ln \mu_l = {\psi_D}^2 \ln \mu_D + {\psi_F}^2 \ln \mu_F + 2\psi_D \psi_F \ln \left(\frac{2\mu_D \mu_F}{\mu_D + \mu_F} \right) \tag{11}$$

in order to simulate a reasonable viscous behavior of the liquid mixture. The surface tension coefficient is instead set to a constant averaged value $\overline{\sigma}$ for the sake of simplicity. Thus, phenomena caused by the Marangoni effect are neglected.

FS3D is parallelized using MPI and OpenMP allowing simulations on supercomputers like the Cray XC40 at the High Performance Center Stuttgart (HLRS), where the simulations presented in this study were calculated. A more detailed overview of further capabilities of FS3D can be found e.g. in Eisenschmidt et al. [5].

3 Computational Setup

The computational setup is shown in Fig. 2a. A drop with diameter D is initialized in a distance of $3D$ above a liquid film of thickness h. In order to increase the spatial grid resolution, only a quarter of the interaction is simulated. The drop exhibits an

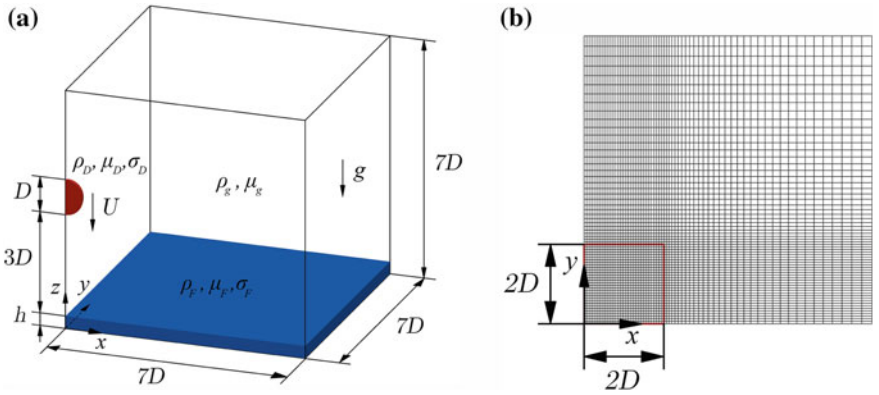


Fig. 2 Computational domain for simulating binary drop film interactions

initial velocity U towards the liquid film, which can also be considered as the impact velocity. The velocity field of the drop is superimposed with a random undirected noise field exhibiting a standard deviation of 3% of the initialized velocity U to trigger instabilities during the interaction. The computational domain with dimensions $7D$ in each direction is discretized with a Cartesian grid. The grid in the cubic impact region with dimensions $2D^3$ is equidistant, whereas outside this region a non-equidistant cell arrangement is used. For both regions the same number of grid cells is used for each direction, meaning that the impact region is discretized with one-eighth of all grid cells used. The cell arrangement is depicted exemplarily in Fig. 2b for the x, y-plane. Besides the two symmetry boundary conditions in the x, z- and y, z-plane, a no-slip boundary condition is applied at the bottom and homogeneous Neumann boundary conditions are used on all remaining sides.

In this study three different impact scenarios were simulated: Deposition, transition and splashing. All three cases were simulated with four different grid resolutions, 128^3, 256^3, 512^3 and 1024^3 cells, respectively, in order to identify the necessary grid resolution for accurately reproducing the expected impact scenario, if possible. The cases were chosen with regard to the splashing threshold proposed by Geppert et al. [9], who used n-hexadecane $C_{16}H_{32}$ and Castrol Hyspin AWS 10 (hydraulic oil) as film and drop liquid, respectively. For this reason, the same physical properties with the ratios $\lambda = 4.580$ and $\chi = 1.136$ are chosen for the liquids in the simulations. The physical properties of these liquids can be found in [9]. The non-dimensional film thickness is set to $\delta = 0.36$ and $\overline{Oh} = 0.04$. Furthermore, the gravitational acceleration is set to $g = 9.81$ m^2/s and the physical properties of the ambient air are set to $\rho_g = 1.204$ kg/m^3 and $\mu_g = 1.813 \times 10^{-5}$ Pas.

For a better understanding, the simulated cases are depicted in Fig. 3 in a $\overline{Oh}$, $\overline{Re}$-diagram showing the impact regimes and the above mentioned correlations, equations (2) and (3), as well. In order to show both correlations in one diagram, the correlation by Geppert et al. [9] (Eq. (2)) was rewritten as follows

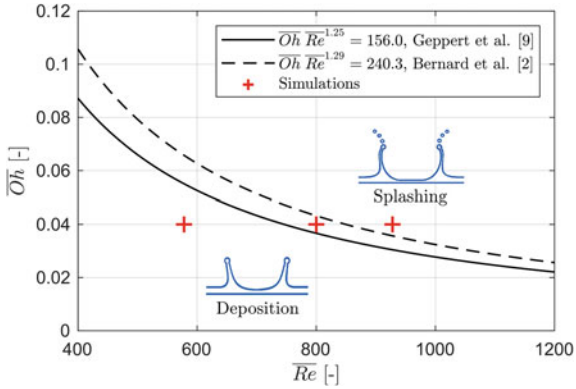


Fig. 3 $\overline{Oh}$, $\overline{Re}$-diagram for mapping of the simulated binary drop film interactions (deposition: $\overline{Re} = 578$, transition: $\overline{Re} = 800$, splashing: $\overline{Re} = 928$) with corresponding impact regimes according to correlations by Geppert et al. [9] and Bernard et al. [2] for $\delta = 0.36$ and $\overline{Oh} = 0.04$

$$\overline{Oh}\,\overline{Re}^{5/4} = \left(114 + 163\delta^{6/5}\right)\left(\frac{\sigma_D}{\overline{\sigma}}\right)^{1/2}\left(\frac{\overline{\rho}}{\rho_D}\right)^{3/4}. \tag{12}$$

As one can see, the transition case is located between both correlations and very close to the splashing threshold. For this reason, the transition case is numerically the most challenging one since no (Bernard et al. [2]) or only few (Geppert et al. [9]) secondary droplets are expected to be generated during the impact. Even though the correct outcome is, therefore, uncertain, we expect to see already secondary droplets since the cases of the experimental campaign by Geppert et al. [9] are reproduced.

4 Results

In the following, we present the grid study of the transition case which lies close to the splashing threshold. After that, the ejected mass of liquid of all simulations and the results using the finest grid resolution are shown. For the evaluation, the non-dimensional time $\tilde{t} = tU/D$ is introduced.

Figure 4 shows the impact morphology of the transition case for all four grid resolutions at $\tilde{t} = 18$. One can see, that for all used grids secondary droplets are generated. The simulations using the two coarsest grids (Fig. 4a–b) show a strong fragmentation of the crown wall with a complete disintegration for the coarsest grid. For the other two grid resolutions (Fig. 4c–d) droplets are already ejected from liquid fingers at the crown rim, reproducing the physical mechanism of droplet generation. However, even if 512 grid cells per dimension are used, an onset of the crown wall fragmentation can be observed, which results in additional secondary droplets. Furthermore, the simulations using the two coarsest grids show also an unphysical rupture of the liquid film inside the crown. This clearly shows that the dimensions of the resulting liquid structures during the impact are too small to be captured by these two computational grids and the numerical method used.

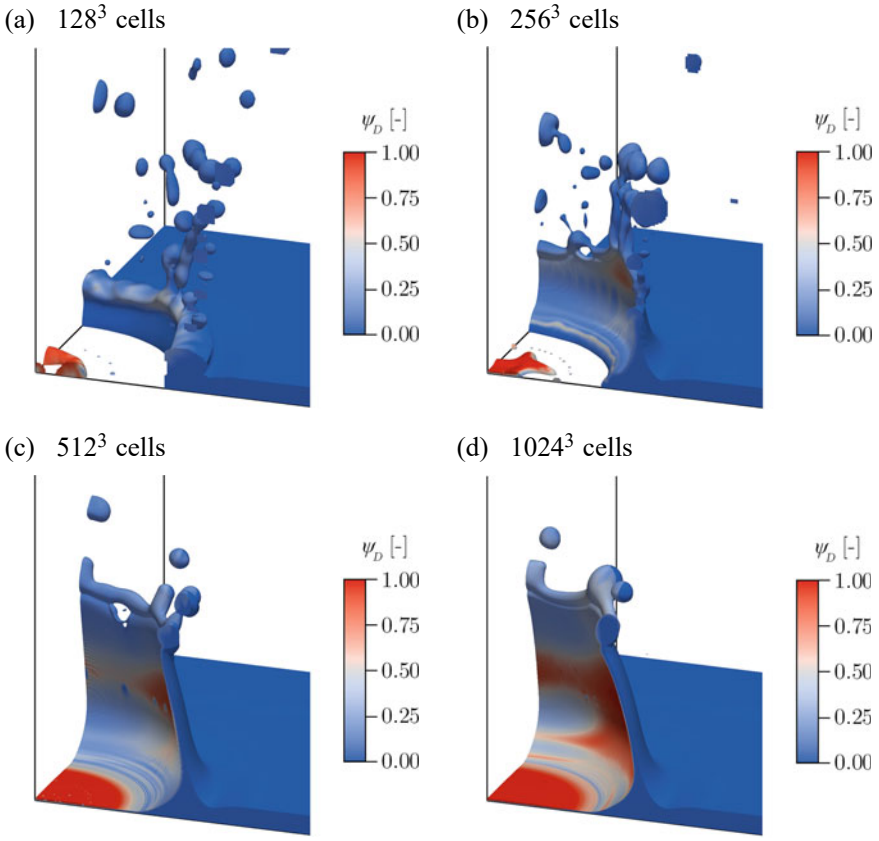


Fig. 4 Impact morphology of the simulated transition case ($\overline{Oh} = 0.04$, $\overline{Re} = 800$, $\delta = 0.36$, $\chi = 1.136$, $\lambda = 4.580$, $\gamma = 1$) at $\tilde{t} = 18$ for different grid resolutions; a) 128^3 cells; b) 256^3 cells; c) 512^3 cells; d) 1024^3 cells

For all simulations also the fraction of drop liquid ψ_D is shown at the liquid surface and additionally in the x, y-plane. As can be seen, the crown wall consists almost only of liquid from the initial film, which is in accordance with an estimation by Gomaa [11]. The reason for this is that the initial spherical drop spreads to a large extent in radial direction. Due to that, only a thin layer of drop liquid contributes to the liquid of the crown wall, as can be seen in Fig. 4c–d. The thickness of this layer often falls short of the minimum distance from the liquid surface in order to ensure an unaffected species advection in the simulation. In such a case, the species profile widens in the vicinity of the liquid surface due to the applied homogeneous Neumann boundary condition at the surface, mimicking an enhanced small-scale mixing between both species. This situation can be seen in many areas on the crown wall and for all grid resolutions. One has to note, that the mixing process and the degree of mixing during two-component drop film interactions with miscible liquids

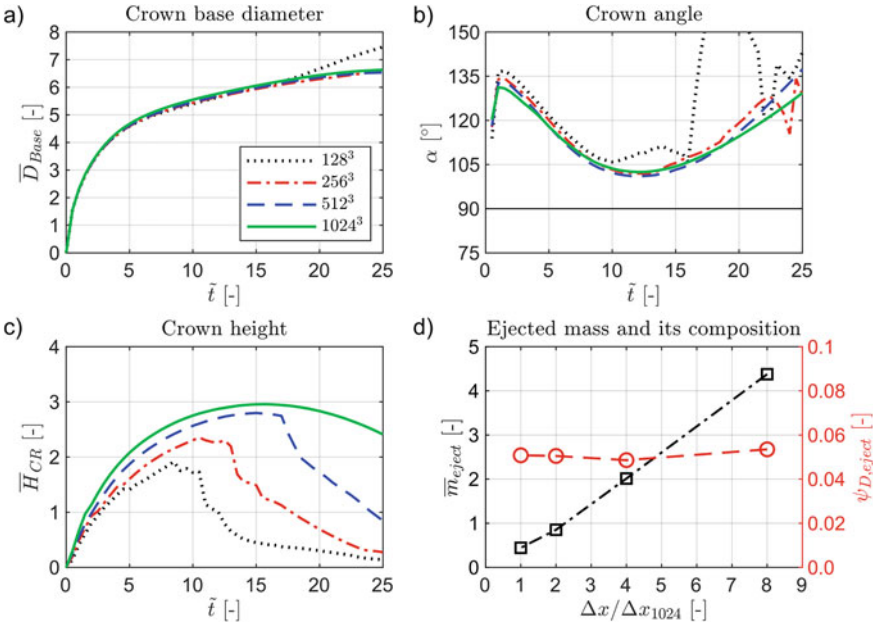


Fig. 5 Temporal evolution of crown parameters and properties of secondary droplets for the transition case ($\overline{Oh} = 0.04$, $\overline{Re} = 800$, $\delta = 0.36$, $\chi = 1.136$, $\lambda = 4.580$, $\gamma = 1$) for different grid resolutions; a) crown base diameter $\overline{D}_{Base}$; b) crown angle α; c) crown height $\overline{H}_{CR}$; d) ejected mass $\overline{m}_{eject}$ and fraction of drop liquid of the ejected mass $\psi_{D,eject}$ over normalized grid cell size

is still an open question and a debated topic in the literature [10]. However, since Geppert showed in [10] already for the here presented transition case phenomena caused by the Marangoni effect (i.e. surface tension gradients exist at the crown wall) the degree of mixing in our simulation using the fine grid can be considered reasonable.

Besides the impact morphology, several parameters were evaluated to see the effect of different grid resolutions on characteristics of the numerically reproduced crown: The non-dimensional crown base diameter $\overline{D}_{Base} = D_{Base}/D$, the crown angle α and the non-dimensional crown height $\overline{H}_{CR} = H_{CR}/D$ (see Fig. 1). The temporal evolution of these three parameters is shown in Fig. 5a–c. The methodology of evaluation is described in Kaufmann et al. [13]. As one can see, already for coarse grid resolutions $\overline{D}_{Base}$ is converged. Only for the coarsest grid the temporal evolution shows a kink at later times, caused by the unphysical rupture of the film inside the crown. An additional grid refinement towards the bottom should, therefore, be sufficient to be able to use the coarsest grid resolution in order to reproduce $\overline{D}_{Base}$ accurately. For simulations with smaller δ a grid resolution of 256 cells per dimension should be used. The crown angle α starts to differ at $\tilde{t} \approx 9$ for the coarsest grid resolution. One reason for this is that the evaluation of the angle is coupled to the crown height $\overline{H}_{CR}$, which shows a sudden drop at a similar point in time (see

Fig. 5c), marking the moment of the unphysical breakup of the crown wall. For all grid resolutions but the finest a drop of the crown height is visible, however, at different points in time. This shows that only by using very fine grid resolutions a stable crown can be simulated. Thus, coarser grid resolutions will shift the splashing threshold to a significant extent to lower impact energies, since a lower impact energy leads to a lower crown height, a thicker crown wall and a potential later breakup time. In order to receive a converged temporal evolution of the crown angle, 256 cells per dimension can be used, whereas for the crown height at least 512 cells in each direction are necessary for the given δ.

Furthermore, Fig. 5d shows the ejected mass normalized with the initial drop mass $\overline{m}_{eject} = m_{eject}/m_D$ and the fraction of drop liquid of that mass $\psi_{D,eject}$. Both parameters are plotted over the grid cell size in the impact region Δx normalized with the cell size of the finest grid Δx_{1024}. From the evaluated masses one can observe that a halved grid cell size leads to the halving of ejected mass and that this trend does not change even for higher grid resolutions. However, the unphysical breakups that take place for all but the finest grid resolution are an additional source of secondary droplets leading to a significant amount of ejected mass. Therefore, we can assume that the amount of ejected mass has stabilized for the simulation with the finest grid due to the fact that no unphysical breakup occurs. A further refinement will not affect the generation of secondary droplets resulting from the disintegration of liquid fingers at the crown rim (see Fig. 4d) caused by the Plateau-Rayleigh instability. Thus, we conclude that this simulated case lies above the splashing threshold, which is in accordance with the correlation by Geppert et al. [9] and is in contrast to the prediction by the correlation by Bernard et al. [2]. Nevertheless, for a confirmation of this conclusion an additional simulation with a grid resolution of 2048 grid cells per dimension would have to be performed. Another interesting result is that the composition of the ejected mass is not influenced by the grid resolution, since it remains nearly constant at 5% (see Fig. 5d). The reason for this is that always the same amount of drop liquid flows into the crown due to the already converged crown base diameter. Therefore, the composition of the secondary droplets generated during two-component drop film interactions can be analyzed in the future by using the same grid resolution as used to receive a converged crown base diameter $\overline{D}_{Base}$.

Figure 6 shows $\overline{m}_{eject}$ of the three simulated drop interactions over the number of grid cells per dimension $N_{c,dim}$. For the deposition case a significant amount of mass is ejected when using the two coarsest grid resolutions. Both resolutions are, therefore, insufficient for reproducing the expected impact outcome. When using the next finer grid resolution, the ejected mass drops almost to zero, indicating that the simulated impact outcome lies in the deposition regime being in accordance with Eq. (2). The splashing case shows a trend similar to the one of the already discussed transition case. For all grid resolutions the amount of ejected mass is higher than that for the transition case by $\approx 0.15 - 0.25$.[1]

[1] Please note that a simple extrapolation of the ejected mass in order to find the splashing threshold does not work. The reason for this is that crossing the threshold results in immediate ejection of several droplets whose total and unknown mass is the lower bound of the possible ejected mass.

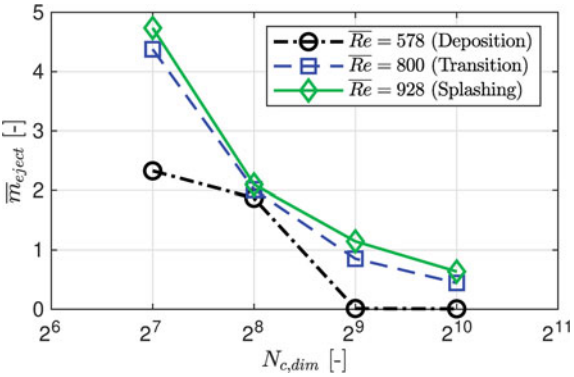


Fig. 6 Ejected mass of the simulated drop film interactions normalized with the initial drop mass for different grid resolutions

The three simulations using the finest grid resolution are shown in Fig. 7 for $\tilde{t} = 5$ and $\approx 2\Delta\tilde{t}$ after the crown has reached its maximum height. A significant distinguishing characteristic between these three interactions is the stability of the crown rim. The rim of the deposition case shows only weak perturbations in form of small azimuthal waves at $\tilde{t} = 5$. A growth of these perturbations, being the root of potential liquid fingers [28], cannot be observed in the further course of the impact (see Fig. 7a at $\tilde{t} = 10$). This indicates that drop film interactions with higher $\overline{Re}$ and, therefore, much closer to the splashing threshold can be performed before secondary droplets are generated. Thus, it is justified to assume that FS3D is capable of reproducing the splashing threshold with sufficient accuracy for two-component-drop film interactions. In contrast to that, small cusps can be seen at $\tilde{t} = 5$ for the transition case, which grow and finally lead to the generation of secondary droplets (Fig. 7b). In the splashing case small liquid fingers can already be observed from the beginning, followed by the continuous generation of secondary droplets (Fig. 7c).

The splashing case reveals a further peculiarity. Tiny bubbles of air entrained during the impact reach the surface in the further course of the interaction. This results in a pointwise change of species composition at the surface (see Fig. 7c at $\tilde{t} = 19$), since the species located beneath the layer of drop liquid is dragged along by the bubble. The resulting surface tension gradient at the surface would explain a pointwise breakup of the crown wall caused by the Marangoni effect, which often can be seen in two-component drop film interactions and also for this particular splashing case [10]. However, this potential breakup mechanism is beyond the scope of the presented study and will be a topic of future investigations.

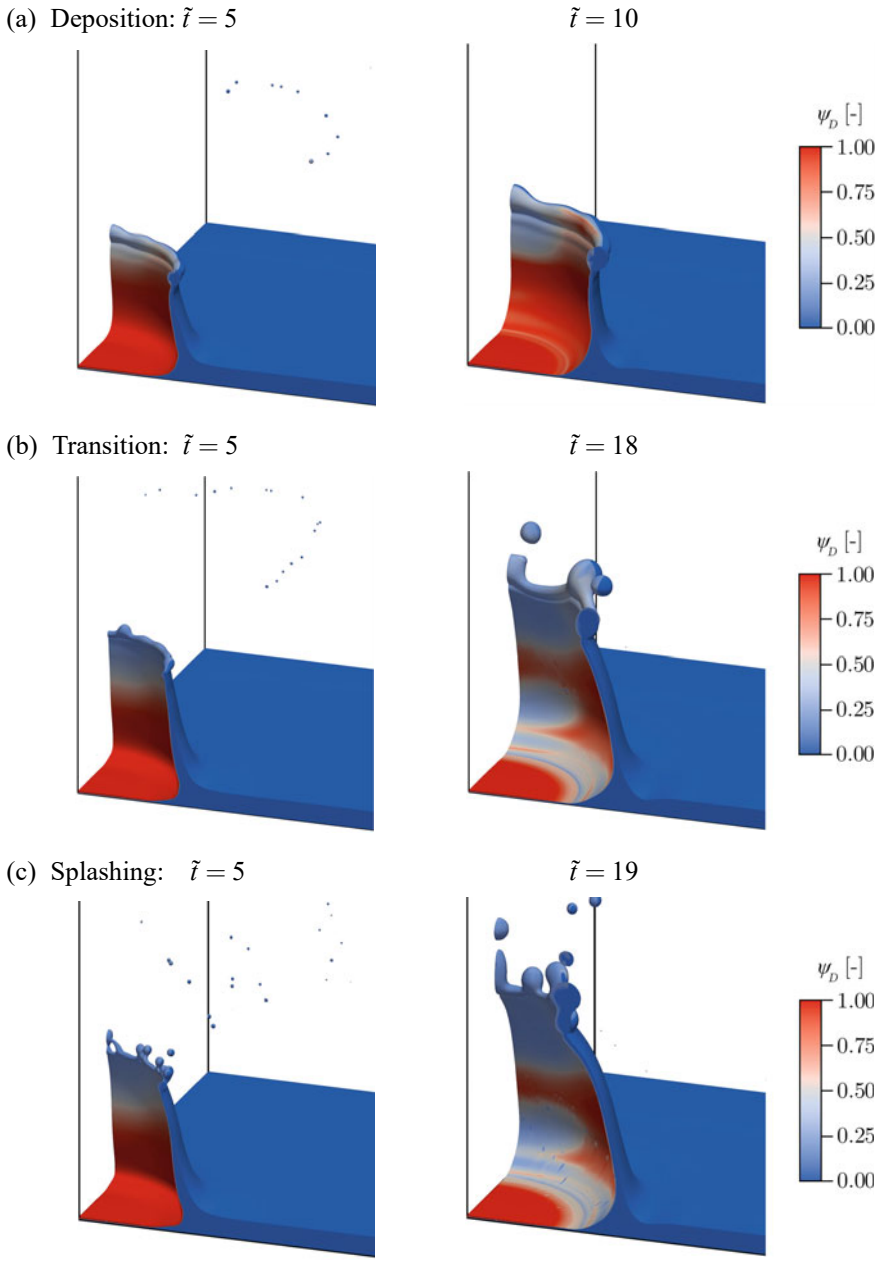


Fig. 7 Impact morphology for the simulated cases at $\tilde{t} = 5$ and $\approx 2\Delta\tilde{t}$ after the crown has reached its maximum height using a grid resolution of 1024^3 cells; a) Deposition; b) Transition case; c) Splashing

5 Computational Performance

5.1 Increasing the Efficiency of the Multigrid Solver

Since the beginning of the development of FS3D one of the main issues is the solution of the pressure Poisson equation. The equation is a result of the volume conservation in incompressible flows and reads

$$\nabla \cdot \left[\frac{1}{\rho(f, \psi_i)} \nabla p \right] = \frac{\nabla \cdot \mathbf{u}}{t}. \tag{13}$$

Rieber [23] first implemented a MG solver into FS3D in order to solve the resulting set of linear equations efficiently. Later, the code was extended and parallelized using MPI and OpenMP, thus, we were able to run simulations on several generations of supercomputers at the HLRS with up to eight billion cells. However, when using multiple processors, the performance of the code decreased dramatically due to the challenging parallelization of the MG solver. Ertl et al. [6] reported the issue and found an unfavorable communication imbalance in the MG solver. They affiliated it to the agglomeration and distribution of the whole computational domain onto one single processor at a certain level during the V-Cycle of the solver. This leads to a very high amount of idling processes, and therefore, to a worsening of performance. As a consequence, Ertl et al. [6] integrated a massively parallel geometric MG solver of the code UG4 [27] in cooperation with the group of Wittum[2] into FS3D. For a high scalability UG4 uses a tree structure for communication, which makes it possible to decrease the solution time for a partial differential equation, such as the pressure Poisson equation [26]. Although the use of UG4 within the FS3D framework showed a generally better performance for strong scaling, especially with a high amount of processors it was still behind expectations. For weak scaling the old MG solver even showed a better performance for a low processor number. However, when using more than 4000 processors, the performance with both solvers collapsed.

The idea in the frame of this study is now to combine the advantages of the two MG solvers. Thus, the old MG solver, which is efficient for weak scaling, was adapted and enhanced taking into account the basic idea of UG4. This is the tree structured communication during the coarsening and refining on the levels during the solution cycle. We changed the way of communication in FS3D at the above mentioned positions where the agglomeration and distribution of the computational domain takes place. Hence, we are now able to gain an efficient communication with the originally implemented MG solver. The benchmark setup as well as the results of a performance analysis are presented in the following sections.

[2] Goethe Center for Scientific Computing (G-CSC) of the Goethe University Frankfurt.

5.2 Benchmark Setup and Performance Analysis

The following section provides a summary of the performance analysis for both FS3D code versions comparing the optimized MG solver with the former originally implemented version. Even though hybrid parallelization is possible (spatial domain composition with MPI and OpenMP on a loop level, see [8]), we used only spatial domain composition with MPI parallelization. This is intended as to exclude the possibility of corrupting the computational performance through multithreading since only the communication and computational performance on the MPI level is desired to be investigated. We measured the scalability and parallel performance using both strong and weak scaling. The number of processors is varied from 2^3 up to 32^3 which directly corresponds to the number of initiated MPI-processes. All calculations were carried out with a duration of 20 min without the output of field data or restart files. We tracked the exact initialization time and employed the effective wall time to estimate the number of calculation cycles per hour (CPH) which is used as a mean to represent the overall performance of the code. As droplets are ejected from the advancing crown in the splashing regime, we used a single isothermal water droplet at standard ambient conditions ($T = 293\,\mathrm{K}$ and pressure of one atmosphere) as a representative benchmark case, similar to the test case presented in [20, 25]. The droplet is set up in a cubic domain with an edge length of 8 mm and initialized as an ellipsoid with semi-principal axis of $a = b = 1.357\,\mathrm{mm}$ and $c = 0.543\,\mathrm{mm}$ to promote droplet oscillation.

5.2.1 Strong Scaling

For the analysis of the strong scaling a baseline case with a grid resolution of 512^3 cells was used while the number of cores was increased successively. The setup is summarized in Table 1 depicting the amount of assigned MPI-processes as well as the resulting amount of cells per process which were allotted to a single processor. Previous studies have shown that for simulations with FS3D on the Cray XC40 system, the ideal number of processors per node is 16. However, for the 32^3 processor setup, which is just an exemplary case, all 24 cores per node were assigned to reduce the required computational resources.

The reference case was calculated on eight cores since memory limitations did not allow allocation on a single compute node. We calculated the strong scaling efficiency (SSE) by relating the amount of estimated calculation cycles per hour with N processors CPH_N to the number of cycles of the reference case CPH_8 (averaged to obtain a comparative single processor value):

$$SSE = \frac{CPH_N}{N/8 \cdot CPH_8} \cdot 100\ \%. \tag{14}$$

Table 1 Strong scaling setup and number of estimated calculation cycles per hour (CPH)

Problem size	512^3				
MPI-processes	2^3	4^3	8^3	16^3	32^3
Cells per process	256^3	128^3	64^3	32^3	16^3
Nodes	2	4	32	256	1366
Processes per node	4	16	16	16	24
CPH Original MG	85	421	2515	507	*n/a*
CPH Optimized MG	88	436	2823	9563	5532

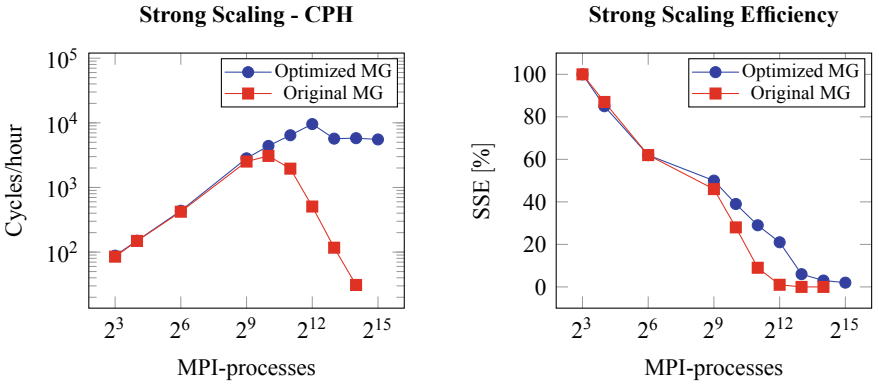


Fig. 8 Number of estimated cycles per hour (on the left) and strong scaling efficiency (on the right) over the number of processes for the optimized and originally implemented MG solver

Note, instead of the classical definition for strong scaling efficiency using the amount of time to complete a predefined work load, we took the number of completed calculation cycles which is equivalent. The results are also summarized in Table 1. Intermediate steps (e.g. $4 \cdot 8^3$ MPI-processes) were taken into account in order to locate the peak performance but are not listed for the sake of clarity.

The estimated CPH as well as the strong scaling efficiency show a similar trend up to 512 processors where the optimized MG solver performs slightly better than the originally implemented version, as depicted in Fig. 8. The peak performance of the original solver is reached at 1024 processors followed by a drastic decrease in both performance and efficiency, especially unfavorable for more than 4000 processors. Results for 32^3 processors for this case are not available as the code could not perform a single calculation cycle during the assigned walltime. However, previous performance studies of FS3D have shown for this case that estimated CPH are much too small to perform an economical simulation.

The performance of the optimized solver peaks at 4096 processors and is about 20 times faster than the version without tree structured communication during the solution cycle, achieving more than 9500 CPH. Even when increasing the number of processors, CPH do not plummet but rather stay constant at approximately

5500 CPH, however, decreasing the strong scaling efficiency progressively. Selecting the case for $4 \cdot 16^3$ processors, almost 190 times more CPH can be executed indicating a massive improvement both in terms of performance and scalability due to the optimized communication structure in the MG solver.

5.2.2 Weak Scaling

For the weak scaling measurements, we kept the amount of cells per processor constant at 64^3 while increasing the number of processors from 2^3 to 32^3, leading to problem sizes with 128^3 to 2048^3 computational cells. The corresponding setups are given in Table 2. The reference case consisted of 128^3 grid cells and was run on eight processors. We estimated the CPH analogously to the strong scaling case for the performance and calculated the parallel efficiency for weak scaling (WSE) given the formula

$$WSE = \frac{CPH_N}{CPH_8} \cdot 100\ \%. \tag{15}$$

The results for both MG versions are also summarized in Table 2. A measurement estimate for the largest case is not available due to the reasons mentioned before. Both code versions exhibit a similar trend for the estimated CPH up to 512 MPI-processes, as can be seen in Fig. 9. The optimized MG solver, however, outperforms the original solver in all cases and achieves performance speed-ups of ≈30% for the case with 64 processors and 15% for the baseline case as well as 512 cores. Calculations with more processors show a drastic decrease in computational performance for the solver in its original state. The optimized communication structure overcomes this issue and leads to a performance increase of almost 10 times the original version at 4096 processors. A slightly stronger decrease in CPH can be observed for the largest case of 32^3 processors where a value of roughly 1800 CPH could have been expected. This might be due to the reduced available bandwidth as all 24 cores per node were assigned to reduce the required computational resources. When we take a look at the weak scaling efficiency, we observe a similar trend where the optimized solver has a better efficiency in general. The original MG solver shows a similar weak scaling efficiency for 512 processors whereas calculations with 4096 or more processors are not feasible anymore. Here, the optimized solver has a weak scaling efficiency of 43%, for the largest problem size it is still 28%.

Note, that the weak scaling efficiency decreases for an increasing number of employed MPI-processes. This stems from the fact of rising numbers of coarsening levels and consequently the varying amount of cycles in the MG solver increasing the communication load due to more point-to-point and global MPI data exchanges. This issue is inevitable for algorithms that employ heavy use of global communication patterns such as the utilized MG solver. Yet, it can be addressed and improved in future performance work in FS3D to a certain extent by exploiting more efficient nearest neighbor communication patterns and further modifying the MG solver especially on the coarsest levels of solving the set of linear equations.

Table 2 Weak scaling setup and number of estimated calculation cycles per hour (CPH)

Cells per process	64^3				
Problem size	128^3	256^3	512^3	1024^3	2048^3
MPI-processes	2^3	4^3	8^3	16^3	32^3
Nodes	2	4	32	256	1366
Processes per node	4	16	16	16	24
CPH Original MG	4690	3161	2544	219	*n/a*
CPH Optimized MG	5335	4163	2977	2286	1482

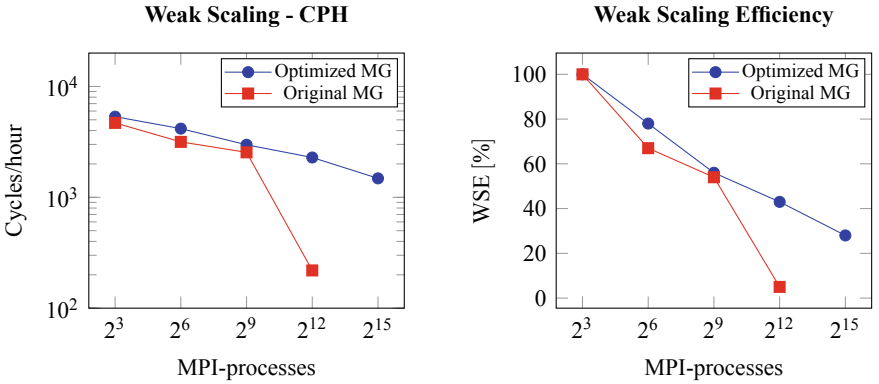


Fig. 9 Number of estimated cycles per hour (on the left) and weak scaling efficiency (on the right) over the number of processes for the optimized and originally implemented MG solver

6 Conclusion

We investigated the suitability of the numerical determination of the splashing threshold for two-component drop film interactions by means of DNS using the multiphase solver FS3D with its improved multi-component extension. Selected interactions across the empirically determined splashing thresholds by Gepper et al. [9] and Bernard et al. [2] were simulated, each with four different grid resolutions. A detailed grid study for a drop film interaction close to the splashing threshold revealed that only by using very fine grid resolutions a premature break up of the crown wall and, thus, the generation of additional unphysical secondary droplets can be prevented, a fact seldom taken into account in the literature. A grid resolution of 512 cells per dimension still showed an unphysical fragmentation of the crown wall which would result in a significant underestimation of the splashing threshold. However, several crown parameters and also the composition of the secondary droplets can already be predicted reliably for coarse grid resolutions that opens up new perspectives for future investigations. From the ejected mass and the resulting impact morphologies of the cases below and above the splashing threshold, we conclude that FS3D is capable of reproducing the splashing threshold for two-component drop film inter-

actions with sufficient accuracy, even for numerically challenging cases where the drop liquid is denser than the liquid of the film. The numerical effort for determining a general splashing threshold correlation would, however, be immense due to the necessary grid resolution.

Furthermore, the overall performance and parallel efficiency was measured and the strong and weak scaling was investigated for both the optimized and originally implemented MG solver with FS3D. The successful implementation of tree structured communication during the solution cycles revealed a speed-up of almost 20 times the original solver at peak performance. Simulations with more than 4000 processors are now feasible without drastic decrease in performance. This is particularly important for simulations with high grid resolutions as required for DNS. At the same time, this enables simulations on new supercomputers like the HPE Apollo (Hawk) using several thousand processors. FS3D shows speed-up for more cores even though parallel efficiency drops. This issue, due to the nature of the MG solver, can be addressed by means of nearest neighbor communication, namely topology mapping, and modifying the solver on the level of solving the sets of linear equations in future efforts.

Acknowledgements The authors kindly acknowledge the *High Performance Computing Center Stuttgart* (HLRS) for support and supply of computational time on the Cray XC40 platform under the Grant No. FS3D/11142. In addition, the authors kindly acknowledge the financial support of the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through the projects WE2549/35-1, GRK 2160/1, 84292822-TRR75, and under Germany's Excellence Strategy-EXC 2075-390740016. We acknowledge the support of the Stuttgart Center for Simulation Science (SimTech).

References

1. M. Baggio, B. Weigand, Numerical simulation of a drop impact on a superhydrophobic surface with a wire. Phys. Fluids **31**(11), 112,107 (2019)
2. R. Bernard, P. Foltyn, A. Geppert, G. Lamanna, B. Weigand, Generalized analysis of the deposition/splashing limit for one- and two-component droplet impacts upon thin films, in *28th ILASS Europe*, Valencia, Spain (2017)
3. S. Copplestone, Numerical investigation of the splashing threshold of mulit-component mixtures. Master's thesis, University of Stuttgart, Institute of Aerospace Thermodynamics, 2013
4. R. Dey, P. Biswas, A novel and effective approach for viscosity prediction of binary and multicomponent liquid mixtures. J. Mol. Liq. **265**, 356–360 (2018)
5. K. Eisenschmidt, M. Ertl, H. Gomaa, C. Kieffer-Roth, C. Meister, P. Rauschenberger, M. Reitzle, K. Schlottke, B. Weigand, Direct numerical simulations for multiphase flows: an overview of the multiphase code FS3D. J. Appl. Math. Comput. **272**(2), 508–517 (2016)
6. M. Ertl, J. Reutzsch, A. Nägel, G. Wittum, B. Weigand, Towards the Implementation of a New Multigrid Solver in the DNS Code FS3D for Simulations of Shear-Thinning Jet Break-Up at Higher Reynolds Numbers, in *High Performance Computing in Science and Engineering 2017*. ed. by W. Nagel, D. Kröner, M. Resch (Springer, Cham, 2017), pp. 269–287. https://doi.org/10.1007/978-3-319-68394-2_16
7. M. Ertl, B. Weigand, Analysis methods for direct numerical simulations of primary breakup of shear-thinning liquid jets. Atom. Sprays **27**(4), 303–317 (2017)

8. C. Galbiati, M. Ertl, S. Tonini, G.E. Cossali, B. Weigand, DNS investigation of the primary breakup in a conical swirled jet, in *High Performance Computing in Science and Engineering 2015*. ed. by W.E. Nägel, D.H. Kröner, M.M. Resch (Springer, Cham, 2016). https://doi.org/10.1007/978-3-319-24633-8_22
9. A. Geppert, H. Gomaa, C. Meister, G. Lamanna, B. Weigand, Droplet wall-film interaction: impact morphology and splashing/deposition boundary of hypsin/n-hexadecane two-component systems, in *26th ILASS Americas*, Portland, OR, USA (2014)
10. A.K. Geppert, Experimental investigation of droplet wall-film interactions of binary systems. Ph.D. thesis, University of Stuttgart (2019)
11. H. Gomaa, Modeling of liquid dynamics in spray laden compressor flows. Ph.D. thesis, University of Stuttgart (2014)
12. C.W. Hirt, B.D. Nichols, Volume of fluid (VOF) method for the dynamics of free boundaries. J. Comput. Phys. **39**(1), 201–225 (1981)
13. J. Kaufmann, A. Geppert, M. Ertl, R. Bernard, V. Vaikuntanathan, G. Lamanna, B. Weigand, Direct numerical simulations of one- and two-component droplet wall-film interactions within the crown-type splashing regime, in *14th ICLASS*, Chicago, IL, USA (2018)
14. B. Lafaurie, C. Nardone, R. Scardovelli, S. Zaleski, G. Zanetti, Modelling merging and fragmentation in multiphase flows with SURFER. J. Comput. Phys. **113**(1), 134–147 (1994)
15. G. Liang, I. Mudawar, Review of mass and momentum interactions during drop impact on a liquid film. Int. J. Heat Mass Transf. **101**, 577–599 (2016)
16. F. Maichle, Numerische Untersuchung und Modellierung von Wandinteraktionen in Zweiphasenströmungen. Ph.D. thesis, University of Stuttgart, 2006
17. C. Mundo, M. Sommerfeld, C. Tropea, Droplet-wall collisions - experimental studies of the deformation and breakup process. Int. J. Multiph. Flow **21**(2), 151–173 (1995)
18. M. Reitzle, C. Kieffer-Roth, H. Garcke, B. Weigand, A volume-of-fluid method for three-dimensional hexagonal solidification processes. J. Comput. Phys. **339**, 356–369 (2017)
19. M. Reitzle, S. Ruberto, R. Stierle, J. Gross, T. Janzen, B. Weigand, Direct numerical simulation of sublimating ice particles. Int. J. Therm. Sci. **145**, 105,953 (2019)
20. J. Reutzsch, M. Ertl, M. Baggio, A. Seck, B. Weigand, Towards a direct numerical simulation of primary jet breakup with evaporation, in *High Performance Computing in Science and Engineering ' 18*. ed. by W.E. Nägel, D.H. Kröner, M.M. Resch (Springer, Cham, 2019). https://doi.org/10.1007/978-3-030-13325-2_15
21. J. Reutzsch, C. Kieffer-Roth, B. Weigand, A consistent method for direct numerical simulation of droplet evaporation. J. Computat. Phys. **413**, 109455 (2020)
22. W.J. Rider, D.B. Kothe, Reconstructing volume tracking. J. Comput. Phys. **141**(2), 112–152 (1998)
23. M. Rieber, Numerische Modellierung der Dynamik freier Grenzflächen in Zweiphasenströmungen. Dissertation, Universität Stuttgart, 2004
24. M. Rieber, A. Frohn, A numerical study on the mechanism of splashing. Int. J. Heat Fluid Flow **20**(5), 455–461 (1999)
25. J. Steigerwald, J. Reutzsch, M. Ibach, M. Baggio, A. Seck, B.K. Haus, B. Weigand, Direct numerical simulation of a wind-generated water wave, in *High Performance Computing in Science and Engineering ' 19*. ed. by W.E. Nägel, D.H. Kröner, M.M. Resch (Springer, Cham, 2021), pp. 325–341. https://doi.org/10.1007/978-3-030-66792-4_22
26. A. Vogel, A. Calotoiu, A. Strubem, S. Reiter, A. Nägel, F. Wolf, G. Wittum, 10,000 performance models per minute – scalability of the UG4 simulation framework, in *Euro-Par 2015*. ed. by J. Träff, S. Hunold, F. Versaci, vol. 9233 (2015), pp. 519–531
27. A. Vogel, S. Reiter, M. Rupp, A. Nägel, G. Wittum, UG4: a novel flexible software system for simulating PDE based models on high performance computers. Comput. Vis. Sci. **16**(4), 165–179 (2013)
28. A.L. Yarin, D.A. Weiss, Impact of drops on solid surfaces: self-similar capillary waves, and splashing as a new type of kinematic discontinuity. J. Fluid Mech. **283**, 141–173 (1995)

Comparison of Two Airfoils for Active Drag Reduction in Turbulent Flow

Marian Albers, Matthias Meinke, and Wolfgang Schröder

Abstract Active flow control of the turbulent flow over two airfoils of different shape is performed with the goal of increasing the overall aerodynamic efficiency. Large-eddy simulations are performed to investigate the impact of spanwise traveling transversal surface waves which are applied to a large percentage of the suction and the pressure side of the DRA2303 and NACA4412 airfoils at a chord based Reynolds number of $\mathrm{Re}_c = 400,000$. The results show a substantial decrease of the total integrated drag together with a slight increase of the lift. A detailed analysis reveals a decrease of the turbulent stresses and an attenuation of the boundary layer growth, resulting in an improvement of the lift-drag ratio for both airfoils.

Keywords Turbulent boundary layer · Drag reduction · Airfoil · Transversal traveling surface wave · Large-Eddy simulation · Active flow control

1 Introduction

Increasing energy efficiency in air travel is one of the keys to reduce global CO_2 emissions. Furthermore, even savings on the order of one percent in fuel consumption can be decisive in the competition between aircraft manufacturers and have a vast impact on airline economics. The drag of slender bodies moving in a fluid at high Reynolds numbers, e.g., aircraft in cruise flight, is to a large part determined by viscous friction, most often due to turbulent boundary layers developing around the wings and the fuselage. Therefore, reducing this friction drag has been of interest for the past decades. Drag reduction techniques can be classified by whether or not additional energy is introduced into the system. A further classification is possible by

M. Albers (✉) · M. Meinke · W. Schröder
Institute of Aerodynamics, RWTH Aachen University, W üllnerstrasse 5a, 52062 Aachen, Germany
e-mail: m.albers@aia.rwth-aachen.de

W. Schröder
JARA Center for Simulation and Data Science, RWTH Aachen University, Seffenter Weg 23, 52074 Aachen, Germany

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_18

considering techniques which delay or prevent transition from laminar to turbulent flow and techniques which alter the state of an already turbulent boundary layer, thereby reducing the viscous drag.

Among passive techniques, i.e., without the input of external energy, the best known approach is streamwise aligned surface protrusions, so-called riblets, which have proven to reduce friction drag in experimental [7] and numerical setups [12]. Moreover, they are one of the few technologies that have demonstrated to work on real aircraft in flight conditions [29]. Other passive techniques include compliant coatings [10] and superhydrophobic surfaces [15].

Active techniques, which require the introduction of external energy into the system, have the advantage of flexiblity of the control parameters, enabling an adjustment of the control to varying operating conditions. An extensive overview of active in-plane drag reduction techniques is given by [22]. Methods like spanwise oscillating walls [17] and streamwise traveling waves of spanwise wall velocity [23] are to be mentioned in this context. Furthermore, traveling transversal surface waves have shown promising results in experimental [19] and numerical studies [1, 4]. In general, these relatively new techniques have been investigated solely in canonical flow setups, e.g., Poisseuille flow and zero-pressure gradient turbulent boundary layer flow. Nevertheless, they are attractive due to the large potential drag reduction and the substantial attentuation of turbulent motion.

Most of the aforementioned methods, both passive and active, focus on reducing viscous drag in turbulent boundary layers. Many more techniques are available for bluff body flow control [9], i.e., control of the total drag of a body consisting of pressure and viscous drag. That is, a more global perspective needs to be taken into account, as the reduction of viscous drag might result in pressure penalties and vice versa [25]. The results of Reynolds-averaged Navier-Stokes simulations of a wing-body configuation in transonic flow with riblet surfaces [21] showed that not only, as is expected, the friction drag is lowered but that the passive technique influences the shock position and decreases also the pressure drag. Banchetti et al. [6] investigated drag reduction via streamwise traveling waves of spanwise wall velocity over a bump in a channel, i.e., a curved wall surface, and found an increase of the drag reduction for the curved geometry compared to the flat wall. A reduction of the skin friction on airfoils using, for instance, steady blowing can result in a thickening of the boundary layer, causing additional pressure drag from the wake, whereas the opposite can happen for steady suction [5]. Hence, a drag reduction technique is desirable which reduces friction drag while also preventing boundary layer growth, as has been shown for instance for body force control [26]. Another promising technique enabling such a favorable combination is spanwise traveling transversal surface waves for which a coupled decrease of pressure and viscous drag has been shown for airfoil flow [2]. We extend this study by considering another airfoil with a distinctively different geometry. A comparison between the effects of the drag reduction method on the flow over the two airfoils is presented to substantiate the applicability of the spanwise-traveling-wave approach to turbulent flow around slender bodies. Note that this contribution is based on the conference proceeding published in [3].

2 Numerical Method

The numerical method is a high resolution large-eddy simulation (LES) solving the filtered compressible unsteady Navier-Stokes equation on curvilinear grids. For the convective fluxes, the advection upstream splitting method (AUSM) is used, while a central scheme is employed for the viscous terms. The temporal integration is performed by an explicit 5-stage Runge-Kutta method of second-order accuracy. An implicit LES model is used, following the approach of monotonically integrated LES (MILES) by [8] such that the AUSM provides for the necessary dissipation for the smallest scales. Investigations by [20] have shown that no additional explicit turbulence model is required. The code has been used and verified for a wide range of flow problems, including turbulent boundary layer flow over moving surfaces [18] and turbulent airfoil flow [2].

3 Computational Setup

We compare the impact of a surface actuation technique on the flow field over two airfoils. The first geometry is a DRA2303 airfoil, designed for transonic flow [11], with a maximum thickness of 14% chord. The findings for this configuration are already discussed in detail in [2]. The second configuration is a NACA4412 airfoil, for which extensive DNS and LES studies exist [16, 28] and also drag reduction setups have been analyzed [5].

The flow around the airfoil is defined in a Cartesian domain defined by $\mathbf{x} = (x, y, z)$ and the corresponding flow velocities are denoted by $\mathbf{u} = (u, v, w)$. The chord of the airfoil is aligned with the x-axis and the spanwise coordinate is z. Periodicity in the spanwise direction is enforced to obtain an infinite span. The density is given by ρ and the pressure is denoted by p. The predictions are performed by solving the Navier-Stokes equations for unsteady compressible flow. The physical domain has an extent of 50 chords in the x- and y-direction and a width of 10% chord which is sufficient to resolve the largest turbulent scales [16]. A C-type curvilinear mesh is used to discretize the physical domain, the resolution in the wall-normal direction in the very near-wall region is $\Delta y^+|_{wall} < 1.6$ on the suction side and $\Delta y^+|_{wall} < 1.0$ on the pressure side with gradual coarsening off the wall. The resolution is $\Delta x^+ < 23.0$ in the wall-tangential direction and $\Delta z^+ < 9.0$ in the spanwise direction. An angle of attack of $\alpha = 5.0°$ for the NACA airfoil and $\alpha = 2.0°$ for the DRA airfoil is prescribed by adjusting the velocity vector of the incoming flow. A numerical tripping [24] at $x/c = 0.1$ on both sides of the airfoil triggers laminar-turbulent transition. The Reynolds number based on the chord length c is $\mathrm{Re}_c = u_\infty c/\nu = 400,000$ such that friction Reynolds numbers of up to $\mathrm{Re}_\tau = u_\tau \delta_{99}/\nu \approx 400$ are achieved. The Mach number is $\mathrm{M} = 0.1$.

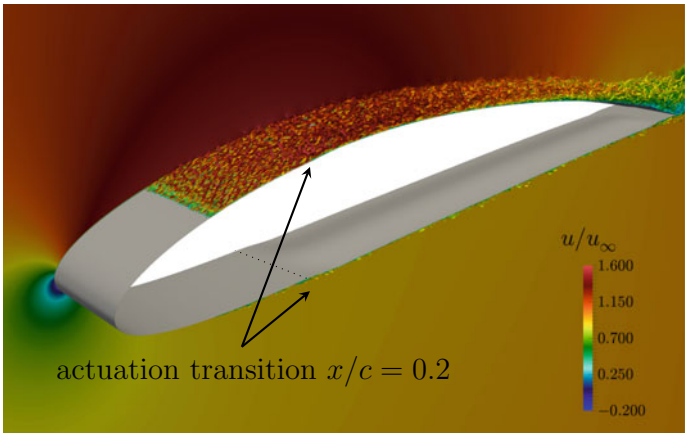


Fig. 1 Contours of the λ_2-criterion of the instantaneous velocity field of the actuated NACA4412 airfoil colored by the instantaneous streamwise velocity

A space- and time-dependent function

$$y_n^+|_{\text{wall}}(x, z^+, t^+) = A^+(x) \cos\left(\frac{2\pi}{\lambda^+} z^+ - \frac{2\pi}{T^+} t^+\right) \tag{1}$$

is applied to control the airfoil surface in the interval $x/c \in [0.2, 0.95]$ on the pressure and suction side such that a sinusoidal deflection of the solid wall in the wall-normal direction traveling in the spanwise direction is obtained. Smooth transitions from a non-actuated to an actuated wall and vice-versa are achieved via $1 - \cos(x)$ functions in the intervals $x/c \in [0.2, 0.25]$ and $x/c \in [0.9, 0.95]$. An illustration of the airfoil and the positions of the onset of the actuation is given in Fig. 1. The distributions of the parameters of the traveling wave function in inner scaling, i.e., the amplitude A^+, the wavelength λ^+, and the period T^+, non-dimensionalized by local values of u_τ, are shown in Fig. 2. A general difficulty in defining a reasonable distribution of the wave parameters is the strongly varying friction velocity along the airfoil chords, especially on the suction side of the NACA airfoil. Therefore, the parameters are chosen to lie in an interval whose bounds are relevant for the current airfoil flow. Based on the knowledge from previous studies the period is kept around a value of $T^+ \approx \mathcal{O}(50)$ (cf. Fig. 2b) and the wavelength is chosen as large as possible, i.e., one wavelength over the whole domain width. A near constant distribution of the amplitude in inner units (cf. Fig. 2c) for the NACA airfoil is achieved by a linear increase of the amplitude in outer scales (cf. Fig. 2d) along the chord on the suction side, on the pressure side a constant value is already adequate. In this regard, the NACA airfoil shows a favorable distribution of the skin-friction, being nearly constant on the pressure side and following a linear decrease on the suction side. In contrast, the distributions around the DRA airfoil create a more challenging situation such that a linear function for the increase of the amplitude in outer scaling does not result in a near-constant distribution in inner scales (cf. Fig. 2d).

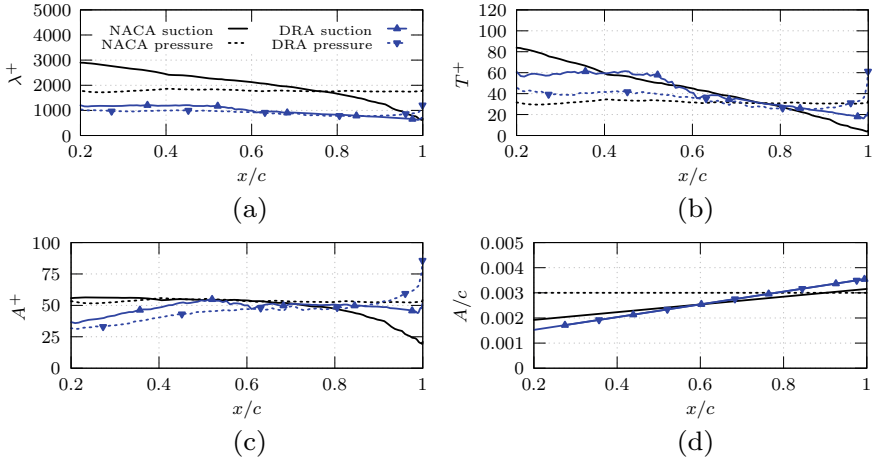


Fig. 2 Traveling transversal wave parameters **a** wavelength, **b** period, and **c** amplitude in inner scaling, and **d** amplitude in outer scaling

Details about the simulation procedure for the DRA airfoil can be found in [2], the simulations for the NACA airfoil are conducted as follows. First, the non-actuated reference setup is run for $tu_\infty/c \approx 24.0$ flow-over times until quasi-steady distributions of the lift and the drag are observed. Then, flow statistics were collected for $tu_\infty/c \approx 7$ flow-over times. Subsequently, the actuated setup is initialized with a converged flow field of the non-actuated reference case and the transition between solid wall and actuated wall is initiated. Once a quasi-steady state is observed statistics are gathered for the actuated flow.

4 Computing Resources

The accurate prediction of the flow field around airfoil wing sections with LES requires to resolve nearly the full spectrum of turbulent motions. That is, the mesh has a near-DNS resolution in the boundary layer and especially close to the wall. Therefore, even for moderate Reynolds numbers several hundred millions mesh points are required. Moreover, due to the small cells in the near-wall region and the explicit time-integration the time step is restricted by the CFL criterion to small values. However, the times until the airfoil flow reaches a stastical steady state are relatively long, since the developing wake flow has an important influence on the pressure distribution around the airfoil. The mesh for the NACA4412 simulation consists of $4698 \times 304 \times 301 \approx 4.3 \cdot 10^8$ points and a multi-block grid partitioning using balanced cut trees [14] is used to achieve a load-balanced distribution over all compute ranks.

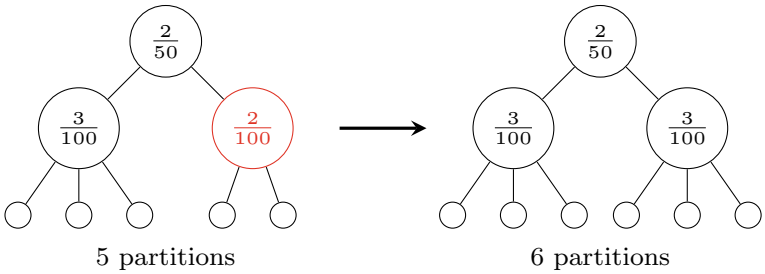


Fig. 3 Example of a balanced cut tree for a two-dimensional computational grid with $N_i \times N_j = 50 \times 100$ cells and (left) five partitions and (right) six partitions; the nodes are labeled with the ratio of children to the number of cells in that particular dimension; the node color red is chosen as an insertion point since the ratio of childs per cells is the lowest among all nodes

That is, for non-multi-block grids a single balanced cut tree is created with the number of tree levels equal to the number of grid dimensions plus one. For instance, as is also shown in Fig. 3, for a two-dimensional grid a cut tree with three levels is created. If only one partition is used the tree will contain only one node on each level. Each additional node on one of the levels below the top level represents a cut along the dimension associated with its level in the tree. The partitioning algorithm then traverses the tree to find an insertion point, i.e., the tree node with the highest ratio of the number of children to the number of cells in that particular dimension. If the selected node for insertion is no leaf node, all sub-trees of this particular node are reset. The sub-trees are filled recursively. The method guarantees that all partitions are themselves structured arrays, thus preserving the full optimization of the numerical code for structured grids. A detailed description of the method and an efficient extension for multi-block grids can be found in [14].

Partitioning of the hexahedral meshes results in n hexahedral sub-partitions, each of which is assigned to one rank. At the partition boundaries the flow variables are exchanged with neighbouring partitions with the message passing interface (MPI) in each Runge-Kutta substep. A strong scaling of the numerical code is shown in Fig. 4. It can be observed that a near optimum scaling is maintained up to 76, 800 cores, i.e., nearly half the available number of cores on Hazel Hen. The simulation of the NACA4412 cases were conducted using 200 compute nodes of the high-performance platform Hazelhen, where each node consists of two Intel® Xeon® E5-2680 v3 CPUs, i.e., a total of 4, 800 cores were used. The non-actuated simulations ran for 5m iterations until a quasi-steady state was observed, the actuated simulation was then restarted from the solution of the reference case and run for another 1.6 million iteration steps, which translates into about 8 flow over times. The simulation of the actuated case ran for about 368 h, i.e., about $1.7 \cdot 10^6$ core hours were used.

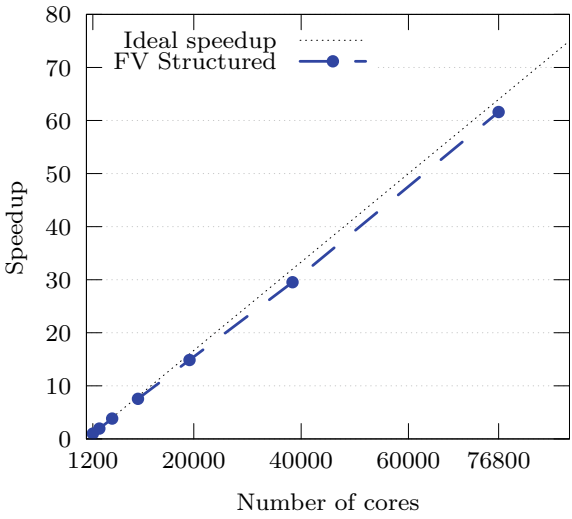


Fig. 4 Strong scaling of the numerical code on Hazel Hen

5 Results

Detailed results on the actuated flow around the DRA2303 airfoil are discussed in [2]. Therefore, statistics from this setup are shown primarily for a comparison with the new results on the NACA4412 airfoil, which we focus on in this study.

An illustration of the actuated flow field of the NACA4412 case is given in Fig. 1. However, no direct conclusion can be derived from the image since the majority of the flow modulation due to the control is confined to the near-wall region. Temporal distributions of the overall drag coefficient c_d, viscous drag coefficient $c_{d,v}$, and lift coefficient c_l for the NACA4412 airfoil are depicted in Fig. 5. Note that all distributions are normalized by the temporal average of the corresponding coefficient of the non-actuated reference case. Similar to the observations in [2], the viscous drag is directly affected after the initialization of the actuation and an averaged decrease of 12.9% is measured. For the DRA2303 case a decrease of 8.6% was achieved. The total drag (cf. Fig. 5b), i.e., the pressure contribution plus the viscous drag, shows the same tendency of being rapidly decreased. On average, it is 8.5% lower compared to the non-actuated reference case, which can be largely attributed to the decrease of the viscous part, whereas the pressure drag is modified by only 3.1%. The short time scale on which the modification of the integrated viscous drag takes place can be explained by the quick development of a periodically fluctuating secondary flow field with wall-normal and spanwise flow velocities above the traveling wave. This flow, resembling a directed oscillating Stokes layer with oscillating fluid instead of an oscillating plate, has an almost immediate effect, i.e., within few periods of the motion [27], on the turbulent structures, especially near the wall. A somewhat larger time scale is apparent for the modification of the lift (cf. Fig. 5c) which is mainly determined by the pressure distribution. Only after about $tu_\infty/c \approx 0.5 - 1.0$

Table 1 Overview of the change of the aerodynamic coefficients in percent of the two airfoils by the traveling wave actuation, negative values indicate an increase.

Case	Δc_d	$\Delta c_{d,v}$	$\Delta c_{d,p}$	Δc_l	$\Delta(L/D)$	ΔA_s
NACA	8.5	12.9	3.1	−1.4	−10.8	−0.5
DRA	7.5	8.6	5.0	−1.4	−9.6	−1.6

flow-over times a departure from the distribution of the non-actuated reference case becomes apparent and a new quasi-steady state is reached not before $tu_\infty/c \approx 4.0$ flow-over times after the onset of the actuation. A similar behavior was observed for the DRA2303 induced through the modified boundary layer and its decreased thickness, altering the flow over the trailing edge and in the wake region and thereby having a delayed effect on the overall pressure distribution. However, it is important to note that on average an increase of the lift by 1.4% is obtained. In combination with the lowered overall drag, this leads to an increase of the aerodynamic efficiency L/D by 10.8%. An overview of all alterations of the aerodynamic coefficients of both airfoils is given in Table 1.

The detailed changes of the skin-friction coefficient c_f are presented in Fig. 6 for the suction and the pressure side. On the suction side of both airfoils, the bulk of the skin-friction decrease is achieved in the forward part of the airfoil, i.e., $x/c < 0.5$, whereas further downstream the effect of the traveling wave actuation is steadily decreasing. This effect is a combination of a reduced efficiency of the control due to the increasing thickness-based Reynolds number of the boundary layer [13] and non-ideal control parameters. Especially the period T, which is constant in outer scaling, is in an efficient range only in a subdomain of the whole streamwise extent of the actuation. On the pressure side of the NACA airfoil, the skin-friction reduction shows a more constant distribution, with a considerable decrease even around $x/c \approx 0.9$. This advantageous behavior can be attributed to the much slower growth of the boundary layer, compared to the suction side, and a nearly constant skin-friction distribution in the reference case. Therefore, the values of the control parameters are in an effective interval over the whole extent of the actuated pressure side. In contrast, the skin-friction distribution on the pressure side of the DRA airfoil shows a decrease similar to the suction side, thus a reduced impact of the actuation can be expected.

A combined analysis of the skin-friction distribution (cf. Fig. 6) together with the momentum thickness distribution shown in Fig. 7 reveals an advantageous feature of this type of flow control. While the skin-friction distribution downstream of the end of the control region quickly recovers and approaches the non-actuated reference state, a persisting effect is evident in the momentum thickness. This was already observed in [26]. The decreased thickness of the actuated boundary layer holds till the trailing edge, resulting in the increased lift and the decreased pressure drag.

To obtain a high overall drag reduction, the streamwise extension of the control region is to be maximized otherwise a skin friction increase due to the decrease of the boundary layer thickness downstream of the control region, as described by

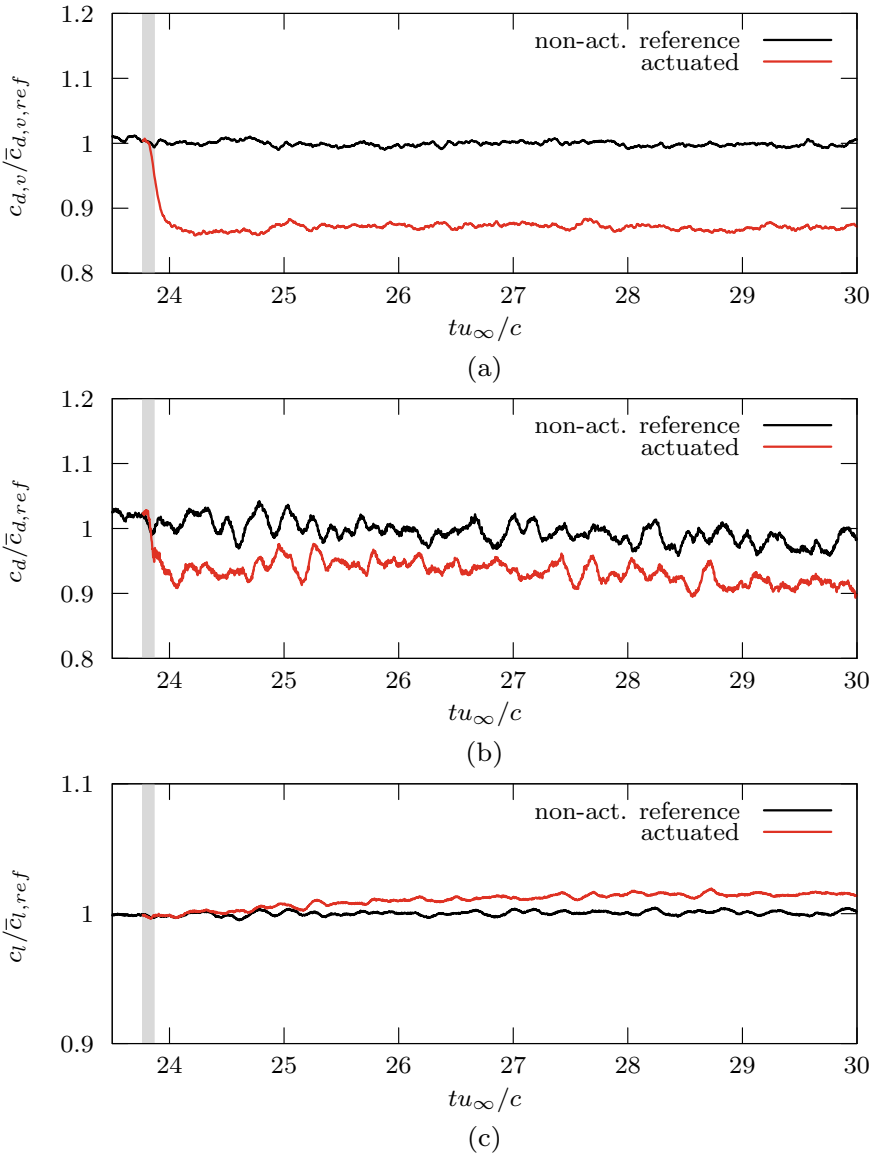


Fig. 5 Temporal distribution of **a** the viscous drag, **b** the total drag, and **c** the total lift of the NACA4412 airfoil of the non-actuated reference case and the actuated case, both normalized by the averaged coefficients of the non-actuated reference case

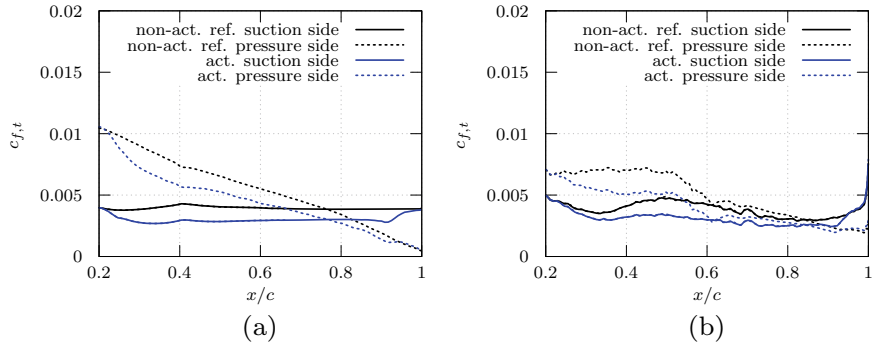


Fig. 6 Temporal and spanwise average of the wall-tangential skin-friction coefficient $c_{f,t}$ of **a** the NACA4412 cases and **b** the DRA2303 cases

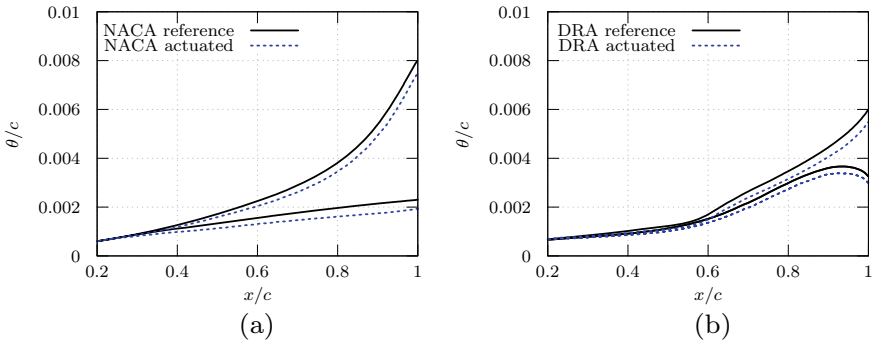


Fig. 7 Temporal and spanwise average of the momentum thickness θ on the suction (solid lines) and pressure side (dashed lines) of **a** the NACA4412 cases and **b** the DRA2303 cases

[26], deteriorates the global measured drag. Therefore, unlike for drag reduction investigations for canonical flows, such as turbulent channel flow or zero pressure gradient turbulent boundary layer flow, multiple additional factors contributing to the total aerodynamic efficiency have to be considered. A global perspective is necessary when evaluating the applicability of any drag reduction method to an airfoil [25]. A closer look at the second-order moments of the velocity and the vorticity is taken in the following. The turbulent velocity fluctuations across the boundary layer height for both airfoils are compared in Fig. 8 on the suction side at the chordwise position $x/c = 0.5$, which corresponds to a friction Reynolds number of $Re_\tau = 312$ for the NACA airfoil and $Re_\tau = 269$ for the DRA airfoil. Although the Reynolds number and the strength of the fluctuations is different at the same streamwise location, the effect of the control is similar. Reductions of all four components are apparent throughout the whole boundary layer. The reductions of the wall-tangential fluctutations are most pronounced. Due to the exceptionally strong attenuation in the near-wall region, i.e., $y^+ < 15$, a direct effect of the actuation on the near-wall streaks can be suspected, similar to spanwise oscillating wall [27]. Furthermore, the strong decrease of the

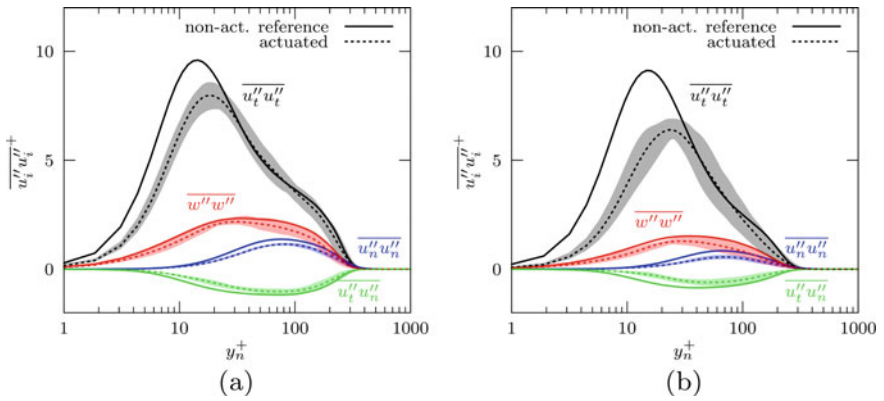


Fig. 8 Wall-normal distributions of the symmetric and the shear-stress components of the Reynolds stress tensor at $x/c = 0.5$ of **a** the NACA4412 cases and **b** the DRA2303 cases, normalized by the friction velocity of the non-actuated reference case. The shaded regions illustrate phasewise variations of the depicted quantity

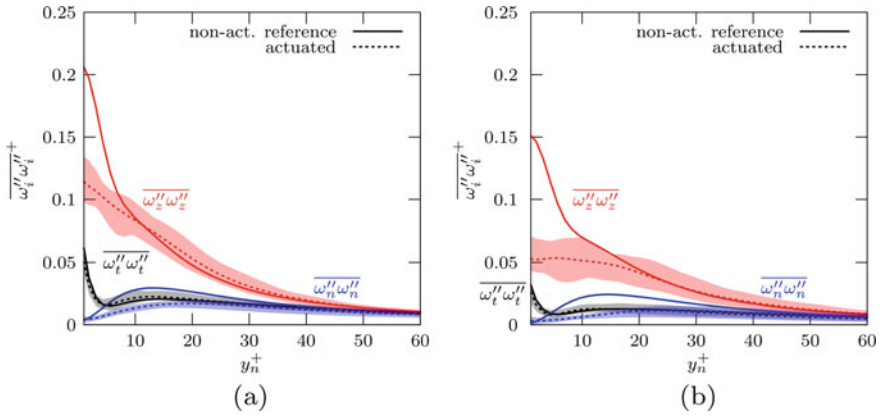


Fig. 9 Wall-normal distributions of the averaged vorticity fluctuations at $x/c = 0.5$ of **a** the NACA4412 cases and **b** the DRA2303 cases, normalized by the friction velocity of the non-actuated reference case. The shaded regions illustrate phasewise variations of the depicted quantity

shear stress, in relative terms, near the wall contributes strongly to the decrease of the skin friction.

This direct interference of the oscillating secondary flow field with the near-wall streaks, in opposition to quasi-streamwise vortices, is corroborated by the distributions of the averaged vorticity fluctuations depicted in Fig. 9. The wall-normal and spanwise vorticity fluctuations close to the wall, which are typically associated with near-wall streaks, are considerably damped. However, there is almost no variation of the wall-tangential component, which would evidence the existence of quasi-streamwise vortices.

6 Conclusion

Large-eddy simulations of turbulent airfoil flow for two airfoil shapes were conducted. The flow control method of spanwise traveling transversal surface waves was applied to the suction and the pressure side of a DRA2303 and of a NAC4412 wing section. The parameters of the time- and space-dependent actuation function were adjusted to be in an efficient range in inner scaling using the local skin-friction.

The results show a general decrease of the integrated viscous drag for both airfoils and also a slight reduction of the pressure drag. In combination with an increase of the lift, owing to a reduced boundary layer thickness at the trailing edge, the aerodynamic efficiency is improved for both airfoil shapes. Based on the current findings it can be stated that, the flow control technique – spanwise transversal surface waves – yields highly promising aerodynamic results for massively different airfoil shapes. Nevertheless, it has to be kept in mind that the airfoil shape and the developing boundary layers require to calibrate the control parameters of the actuation system, especially when a large percentage of the airfoil surface is subject to the control. Although the findings of this study are very promising with respect to active flow control and drag reduction, more details have to be investigated, e.g., higher Reynolds number flows, transonic flows with shocks, etc. are to be analyzed, to assess whether or not drag reduction can be achieved in cruise flight.

Acknowledgment Excerpts of this contribution have previously been published in the TSFP-11 conference proceedings [3]. The research was funded by the Deutsche Forschungsgemeinschaft (DFG) in the framework of the research projects SCHR 309/52 and SCHR 309/68. The authors gratefully acknowledge the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this project by providing computing time on the GCS Supercomputers Hazel Hen at HLRS Stuttgart.

References

1. M. Albers, P.S. Meysonnat, D. Fernex, R. Semaan, B.R. Noack, W. Schröder, Drag reduction and energy saving by spanwise traveling transversal surface waves for flat plate flow. Flow Turb. Combust. **105**, 125–157 (2020). https://doi.org/10.1007/s10494-020-00110-8
2. M. Albers, P.S. Meysonnat, W. Schröder, Actively reduced airfoil drag by transversal surface waves. Flow Turbul. Combust. **102**(4), 865–886 (2019). https://doi.org/10.1007/s10494-018-9998-z
3. M. Albers, P.S. Meysonnat, W. Schröder, Influence of airfoil shape on active drag reduction efficiency. In: International Symposium on Turbulence & Shear Flow Phenomena (TSFP-11) (2019)
4. M. Albers, W. Schröder, Drag reduction for swept flat plate flow. Phys. Rev. Fluids **5**, 064611 (2020). https://doi.org/10.1103/PhysRevFluids.5.064611
5. M. Atzori, R. Vinuesa, G. Fahland, A. Stroh, D. Gatti, B. Frohnapfel, P. Schlatter, Aerodynamic effects of uniform blowing and suction on a NACA4412 airfoil. Flow Turb. Combust. **105**, 735–759 (2020). https://doi.org/10.1007/s10494-020-00135-z
6. J. Banchetti, P. Luchini, M. Quadrio, Turbulent drag reduction over curved walls. J. Fluid Mech. **896**, A10 (2020). https://doi.org/10.1017/jfm.2020.338

7. D.W. Bechert, M. Bruse, W. Hage, J.G.T.V.D. Hoeven, G. Hoppe, Experiments on drag-reducing surfaces and their optimization with an adjustable geometry. J. Fluid Mech. **338**, 59–87 (1997). https://doi.org/10.1017/S0022112096004673
8. J.P. Boris, F.F. Grinstein, E.S. Oran, R.L. Kolbe, New insights into large eddy simulation. Fluid Dyn. Res. **10**(4–6), 199–228 (1992). https://doi.org/10.1016/0169-5983(92)90023-P
9. H. Choi, W.P. Jeon, J. Kim, Control of flow over a bluff body. Ann. Rev. Fluid Mech. **40**(1), 113–139 (2008). https://doi.org/10.1146/annurev.fluid.39.050905.110149
10. K.S. Choi, X. Yang, B.R. Clayton, E.J. Glover, M. Atlar, B.N. Semenov, V.M. Kulik, Turbulent drag reduction using compliant surfaces. Proc. R. Soc. London Ser. A **453**(1965), 2229–2240 (1997). https://doi.org/10.1098/rspa.1997.0119
11. J.L. Fulker, M.J. Simmons, *An Experimental Investigation of Passive Shock/Boundary-Layer Control on an Aerofoil* (Vieweg+Teubner Verlag, Wiesbaden, 1997), pp. 379–400. https://doi.org/10.1007/978-3-322-90711-0_22
12. R. García-Mayoral, J. Jiménez, Drag reduction by riblets. Philos. Trans. R. Soc. London Ser. A **369**(1940), 1412–1427 (2011). https://doi.org/10.1098/rsta.2010.0359
13. D. Gatti, M. Quadrio, Reynolds-number dependence of turbulent skin-friction drag reduction induced by spanwise forcing. J. Fluid Mech. **802**, 553–582 (2016). https://doi.org/10.1017/jfm.2016.485
14. G. Geiser, W. Schröder, Structured multi-block grid partitioning using balanced cut trees. J. Parallel Distrib. Comput. **138**, 139–152 (2020). https://doi.org/10.1016/j.jpdc.2019.12.010
15. J.W. Gose, K. Golovin, M. Boban, J.M. Mabry, A. Tuteja, M. Perlin, S.L. Ceccio, Characterization of superhydrophobic surfaces for drag reduction in turbulent flow. J. Fluid Mech. **845**, 560–580 (2018). https://doi.org/10.1017/jfm.2018.210
16. S.M. Hosseini, R. Vinuesa, P. Schlatter, A. Hanifi, D.S. Henningson, Direct numerical simulation of the flow around a wing section at moderate Reynolds number. Int. J. Heat Fluid Flow **61**, 117–128 (2016). https://doi.org/10.1016/j.ijheatfluidflow.2016.02.001
17. W.J. Jung, N. Mangiavacchi, R. Akhavan, Suppression of turbulence in wall-bounded flows by high-frequency spanwise oscillations. Phys. Fluids A **4**(8), 1605–1607 (1992). https://doi.org/10.1063/1.858381
18. S. Klumpp, M. Meinke, W. Schröder, Drag reduction by spanwise transversal surface waves. J. Turbul. **11**, N11 (2010). https://doi.org/10.1080/14685248.2010.494606
19. W. Li, D. Roggenkamp, V. Paakkari, M. Klaas, J. Soria, W. Schröder, Analysis of a drag reduced flat plate turbulent boundary layer via uniform momentum zones. Aerosp. Sci. Technol. **96**, 105552 (2020). https://doi.org/10.1016/j.ast.2019.105552
20. M. Meinke, W. Schröder, E. Krause, T. Rister, A comparison of second- and sixth-order methods for large-eddy simulations. Comput. Fluids **31**(4), 695–718 (2002). https://doi.org/10.1016/S0045-7930(01)00073-1
21. B. Mele, R. Tognaccini, P. Catalano, Performance assessment of a transonic wing-body configuration with riblets installed. J. Aircraft **53**(1), 129–140 (2016). https://doi.org/10.2514/1.C033220
22. M. Quadrio, Drag reduction in turbulent boundary layers by in-plane wall motion. Philos. Trans. R. Soc. London Ser. A **369**(1940), 1428–1442 (2011). https://doi.org/10.1098/rsta.2010.0366
23. M. Quadrio, P. Ricco, C. Viotti, Streamwise-travelling waves of spanwise wall velocity for turbulent drag reduction. J. Fluid Mech. **627**, 161 (2009). https://doi.org/10.1017/S0022112009006077
24. P. Schlatter, R. Örlü, Turbulent boundary layers at moderate Reynolds numbers: inflow length and tripping effects. J. Fluid Mech. **710**, 5–34 (2012). https://doi.org/10.1017/jfm.2012.324
25. P.R. Spalart, J.D. McLean, Drag reduction: enticing turbulence, and then an industry. Philos. Trans. R. Soc. London Ser. A **369**(1940), 1556–1569 (2011). https://doi.org/10.1098/rsta.2010.0369
26. A. Stroh, Y. Hasegawa, P. Schlatter, B. Frohnapfel, Global effect of local skin friction drag reduction in spatially developing turbulent boundary layer. J. Fluid Mech. **805**, 303–321 (2016). https://doi.org/10.1017/jfm.2016.545

27. E. Touber, M.A. Leschziner, Near-wall streak modification by spanwise oscillatory wall motion and drag-reduction mechanisms. J. Fluid Mech. **693**, 150–200 (2012). https://doi.org/10.1017/jfm.2011.507
28. R. Vinuesa, P.S. Negi, M. Atzori, A. Hanifi, D.S. Henningson, P. Schlatter, Turbulent boundary layers around wing sections up to $Re_c = 1,000,000$. Int. J. Heat Fluid Flow **72**, 86–99 (2018). https://doi.org/10.1016/j.ijheatfluidflow.2018.04.017
29. M.J. Walsh, W.L. Sellers III., C.B. Mcginley, Riblet drag at flight conditions. J. Aircraft **26**(6), 570–575 (1989). https://doi.org/10.2514/6.1988-2554

Application of a JPEG 2000-Based Data Compression Algorithm to DNS of Compressible Turbulent Boundary Layers Up to $Re_\theta = 6600$

Christoph Wenzel, Patrick Vogler, Johannes M. F. Peter, Markus J. Kloker, and Ulrich Rist

Abstract This paper presents a performance analysis of the compression library *BigWhoop* applied to compressible-turbulent-boundary-layer data obtained by direct numerical simulation (DNS). The DNS data are computed for a Mach number of $M = 2.0$ and cover a Reynolds-number range of $Re_\theta = 300$ to 6600. Including six flow-field variables, each time step has a data size of about 1 TB. Mainly evaluated for the streamwise velocity component, various compression rates are tested for the flow field of one single time step as data post processing. It is shown that DNS data compressed up to 1 : 200 yield a maximum absolute error of 0.2% for the entire domain. This is still accurate enough to reliably perform flow-field investigations in the outer layer of the turbulent boundary layer, notably considering that the maximum error only occurs with extremely low probability. The no-slip wall boundary condition forces the velocity down to zero, and thus the relative error, i.e. the absolute error divided by the local velocity, is significantly larger in the near-wall region. Hence, for flow-field investigations depending on important wall values like the skin friction for instance, the maximum compression ratio should be limited to about 1 : 100, keeping the maximum relative error below 1% in the near-wall region.

C. Wenzel (✉) · J. M. F. Peter · M. J. Kloker · U. Rist
Institut für Aero- und Gasdynamik (IAG), Universität Stuttgart, Pfaffenwaldring 21, 70550 Stuttgart, Germany
e-mail: wenzel@iag.uni-stuttgart.de

J. M. F. Peter
e-mail: johannes.peter@iag.uni-stuttgart.de

M. J. Kloker
e-mail: kloker@iag.uni-stuttgart.de

U. Rist
e-mail: rist@iag.uni-stuttgart.de

P. Vogler
High Performance Computing Center (HLRS), Stuttgart, Nobelstraße 19, 70569 Stuttgart, Germany
e-mail: hpcpvogl@hlrs.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_19

1 Introduction

In modern high-performance computing (HPC) systems, the computing power is rapidly growing according to Moore's law. Since the data transfer rates between data storage media are growing more slowly, however, an ever-increasing discrepancy between data input and output is foreseeable for very large simulations in the future, cf. [25].

Current State in DNS of TBLs. An essential point in the investigation of turbulent boundary layers (TBLs) is their rough classification into an inner part close to the wall and an outer part distant from the wall. At low Reynolds numbers ($Re_\theta \approx 1000$, where θ stands for the momentum thickness) both regions strongly interact with each other and the maximum of the turbulent kinetic energy production lies in the transition layer between the viscous sublayer and the logarithmic law-of-the-wall at approximately $y^+ = 12$, where superscript $+$ denotes inner scaling. At high Reynolds numbers, in contrast, the majority of the turbulent kinetic energy is produced in the region of the logarithmic layer. Depending on the respective argumentation, the crossover point at which the production in the area of the logarithmic wall law exceeds that in the transition layer for the first time is assumed at a Reynolds number of approx. $Re_\tau = 4200$ ($Re_\theta \approx 18000$), see [22].

For the investigation of TBLs via DNS the Reynolds number range investigated is typically limited to about $Re_\theta \approx 3000$, in salient cases to a maximum of $Re_\theta = 4300$ (incompressible) and $Re_\theta = 6000$ (compressible). Even if these Reynolds numbers already extend into the range of scale separation and thus can no longer be called pure "low-Reynolds-number solutions", they are still far away from the crossover Reynolds number. As a consequence of the present inaccessibility of the crossover point, the computing power additionally gained in recent years was mainly used to investigate more complex geometries by means of DNS (or LES), since an extension of the computing range to e.g. $Re_\theta = 8000$ would not have led to a noticeable gain in knowledge compared to previous DNS [7].

The first well-documented DNS of a TBL by Spalart [23] at $Re_\theta = 1400$ with $(432 \times 80 \times 320)$ grid points in the three spatial directions and thus a total of about 1.1 million numerical grid points, was the starting point. Under the assumption that the available computing power has increased by a factor of $2^{16} = 65536$ according to Moore's empirical law in the last 32 years since Spalart's DNS, grids with up to 72 billion points should be usable today. Based on the data published in Wenzel et al. [31] and the set-up used therein, this would correspond to simulations on the former Cray supercomputer "Hazel Hen" of the HLRS (High Performance Computing Centre of the University of Stuttgart) of $(30000 \times 750 \times 3200) = 72$ billion grid points and thus a maximum achievable Reynolds number of $Re_\theta \approx 15000$. On the new system HPE-Apollo "Hawk", which provides a performance increase by a factor of about three compared to the Cray system, computational grids with about $(42500 \times 980 \times 4800) = 200$ billion grid points are realizable, which would allow for a maximum Reynolds number of about $Re_\theta = 20000$. For six flow variables being stored in double precision

(8 byte) in our code, this would lead to a file size of 9596 GB (8937 GiB) per time step.

Data Storage Issues. When evaluating DNS data in the past (and often also today), mainly temporally and spanwise averaged flow values were of primary interest, which could be used as reference for turbulence models or analytical models derived from averaged flow values. Based on this research, today's interest is mostly focused on transient effects, where the origin and behaviour of turbulent structures is investigated. From a data storage point of view, the calculation of mean values is no major problem, as it can be performed on-the-fly during a calculation and therefore only generates a single output file. Since investigations of the transient flow field, in contrast, are usually much more complex and often too complicated for an on-the-fly evaluation, a large number of time steps have to be stored, which then have to be investigated separately in parallelized post-processing algorithms. It is therefore evident that the investigation of transient effects in turbulent boundary layers will be extremely limited by the available storage space and the handling of the data to be stored, if the available computing power should be fully utilized to reach the next milestone of $Re_\theta = 20000$.

Objectives of the Study. The main goal of this study is to evaluate the applicability of the compression library *BigWhoop* to compressible-turbulent-boundary-layer data gained by DNS. The DNS data are computed for a Mach number of $M = 2.0$ using the code *NS3D* and cover a Reynolds-number range of $Re_\theta = 300$ to 6600, yielding a data size of about 1 TB for each time step including six flow-field variables. Mainly evaluated for the streamwise component of the velocity field, various compression rates are tested and the errors made evaluated.

This study is structured as follows: At first the main aspects of the DNS are introduced, followed by a short description of the compression algorithm used. Afterwards, the errors of compressed and decompressed DNS data are analyzed and discussed for various compression rates. Finally, the main results are summarized.

2 Numerical Set-Up

Numerical Methodology. The computation is performed with the compressible high-order in-house DNS code $NS3D$ at a free-stream Mach number of $M_\infty = 2.0$. Fundamentals of the code are described in Babucke [3], Linn & Kloker [16, 17] and Keller and Kloker [11, 12]. It solves the three-dimensional, unsteady, compressible Navier-Stokes equations together with the continuity and energy equation in conservative formulation. Assuming a velocity vector $\mathbf{u} = [u, v, w]^T$ in streamwise, wall-normal and spanwise directions x, y and z, respectively, the dimensionless solution vector is $\mathbf{Q} = [\rho, \rho u, \rho v, \rho w, E]^T$, where ρ and E are the density and the total energy. Velocities and length scales are normalized by the dimensional streamwise reference velocity U^*_∞ and the reference length L^*, respectively. Thermodynamical

Table 1 Summary of boundary-layer parameters evaluated at six Reynolds numbers $Re_\theta = \rho_e u_e \theta / \mu_e = 1000 - 6000$. Given parameters are the Reynolds numbers $Re_\tau = \overline{\rho}_w u_\tau \delta_{99} / \overline{\mu}_w$, $Re_{\delta_{99}} = \rho_e u_e \delta_{99} / \mu_e$, $Re_{\delta^*} = \rho_e u_e \delta^* / \mu_e$, the shape factor $H = \delta^* / \theta$ and the skin-friction coefficient $c_f = 2\overline{\tau}_w / (\overline{\rho}_w u_e^2)$

Re_θ	Re_τ	$Re_{\delta_{99}}$	Re_{δ^*}	H	$c_f \times 10^3$
1000	229	10973	3310	3.310	3.270
2000	408	21541	6492	3.246	2.694
3000	580	32240	9639	3.213	2.425
4000	745	43003	12785	3.196	2.246
5000	909	53749	15904	3.181	2.138
6000	1072	64815	19031	3.172	2.043

quantities are normalised with the reference temperature T_∞^* and the reference density ρ_∞^*. Pressure p is non-dimensionalised by $\rho_\infty^* U_\infty^{*2}$. The specific heats c_p and c_v as well as the Prandtl number Pr are assumed to be constant. Temperature dependence of the viscosity is modeled by Sutherland's law. The equations are solved on a block-structured Cartesian grid spanning a rectangular integration domain. All three spatial directions are discretized using 6^{th}-order subdomain-compact finite differences [10] for the present simulations. The classical fourth-order Runge-Kutta scheme is used for time integration.

Boundary Conditions. At the solid wall, the flow field is treated as fully adiabatic with $(dT/dy)_w = 0$ which suppresses any heat exchange between wall and fluid, whereas the pressure at the wall is calculated by $(dp/dy)_w = 0$ from the interior field. For the velocity components, a no-slip boundary condition is applied. The flow field is periodic in the spanwise direction. At the inlet a digital-filtering synthetic eddy method is used to generate an unsteady, pseudo-turbulent inflow condition [13, 29, 31]. In order to prevent the far-field flow from being distorted by this transition process, a sponge region [14] is applied in the inlet region of the simulation domain, which damps down the flow to an unperturbed free-stream baseflow. At the top of the domain, all flow variables are computed such that the gradient along spatial characteristics is zero, except for the pressure, which is derived from the equation of state. At the very outflow, a spatially parabolized outflow boundary condition is applied. The spatial parabolization is achieved by neglecting the second x-derivatives; biased difference stencils are used for the first x-derivative, see [29]. In both wall-normal and streamwise direction, the numerical grid is stretched and the solution filtered [5, 28] in order to avoid reflections from the boundaries. More details about the set-up can be found in [29].

Simulation Parameters. The main region of the computational box has a dimension of $1700\,\delta_{99,0}^* \times 68\,\delta_{99,0}^* \times 24\pi\,\delta_{99,0}^*$ in streamwise, wall-normal and spanwise direction, respectively, where $\delta_{99,0}^*$ is the boundary-layer thickness at the inlet of the simulation domain. It is designed with a height of at least three boundary-layer thicknesses $\delta_{99,end}^*$ and width of about $\pi\,\delta_{99,end}^*$ in spanwise direction, measured at the end

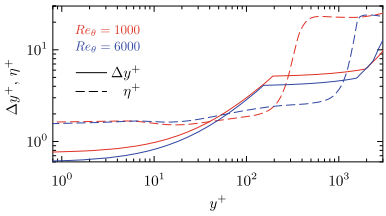


Fig. 1 Resolution check of the numerical grid. The wall-normal resolution $\Delta y^+ = \Delta y u_\tau / \nu_w$ in comparison to the local Kolmogorov scale $\eta^+ = (\nu^3/\epsilon)^{1/4}(u_\tau/\nu_w)$, evaluated at $Re_\theta = 1000$ and 6000

of the main region x^*_{end}. With the momentum-thickness and skin-friction Reynolds numbers $Re_\theta = \rho_e u_e \theta / \mu_e$ and $Re_\tau = \overline{\rho}_w u_\tau \delta_{99} / \overline{\mu}_w$, respectively, the domain covers a usable Reynolds-number range of $Re_\theta = 680 - 6600$ and $Re_\tau = 160 - 1180$, see Table 1. The numerical grid consists of $17000 \times 620 \times 1920$ grid points for the main region and $17200 \times 620 \times 1920$ grid points for the overall domain resulting in 20.474.880.000 total grid points. Calculated in viscous wall units, this gives a grid spacing of $\Delta x^+ = 6.2$, $\Delta y_1^+ = 0.76$ and $\Delta z^+ = 2.4$ at $Re_\theta = 1000$ and of $\Delta x^+ = 4.9$, $\Delta y_1^+ = 0.6$ and $\Delta z^+ = 1.9$ at $Re_\theta = 6000$. A comparison of the wall-normal resolution $\Delta y^+ = \Delta y u_\tau / \nu_w$ with the corresponding Kolmogorov scales $\eta^+ = (\nu^3/\epsilon)^{1/4}(u_\tau/\nu_w)$ is shown in Fig. 1. $\epsilon = \overline{\tau'_{ik} \partial u'_i / \partial x_k}$ is the density-related turbulent dissipation rate. The grid resolution is well below the local Kolmogorov length scale except for a limited y-interval in the layer, see Fig. 1, and thus high enough compared to similar studies. Consequently, a sufficient number of grid points are used in the wall-normal direction. The averaging time after the initial transient is about $\Delta t u_e / \delta_{99} \approx 700$ at $Re_\theta = 1000$ and 120 at $Re_\theta = 6000$. The basic thermodynamic flow parameters are $T^*_\infty = 288.15\,\mathrm{K}$ and $\rho^*_\infty = 1.225\,\mathrm{kg/m^3}$. The Prandtl number is $Pr = 0.71$, the specific gas constant $R^* = 287\,\mathrm{J/kg/K}$ and the ratio of specific heats $\kappa = 1.4$.

3 DNS Performance Aspects

The simulation presented has been carried out on the Cray XC40 'Hazel Hen' supercomputer at HLRS.

The simulation code NS3D is parallelized using a hybrid MPI/OpenMP approach. For the version used in this investigation, the computational grid can be split in the x- and y-direction using MPI for inter-domain communication and within a domain the z-direction is parallelized using the shared-memory OpenMP library. Recently, scaling aspects of this approach have been investigated and some shortcomings have been found [6]. The number of cores used for the z-direction is limited to the 24 available cores per node (then with one MPI process per node), independent of the number of grid points. Additionally, the code has been found to scale well up to 12 OpenMP threads with the parallel efficiency dropping to only around 55% for 24 threads due to socket-remote memory access in the ccNUMA architecture of the Cray XC40. This has been taken as incentive to invest time into improving the code

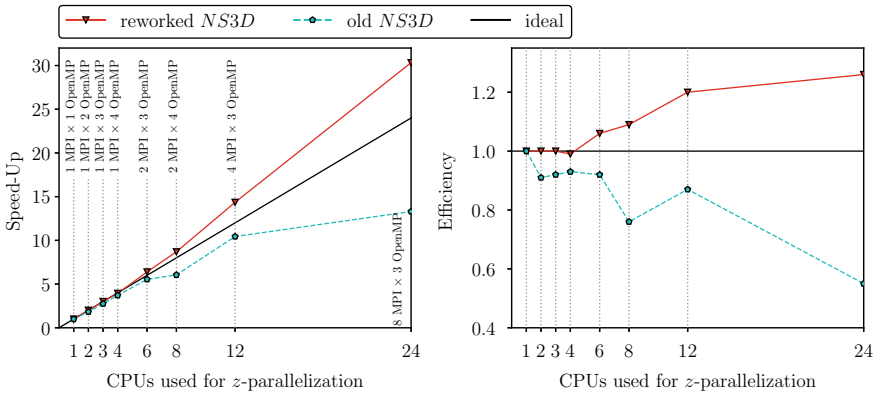


Fig. 2 (Left) Speed-up and (right) efficiency for a variation of the number of CPUs used for the z-parallelization. Scaling is done using pure OpenMP for the old NS3D and a mixture of MPI/OpenMP for the reworked code

performance and scalability to very high core counts by implementing a full MPI decomposition in all spatial directions.

Figure 2 shows the speed-up and efficiency as a function of the CPUs used for the parallelization of the z-direction for the old code version (pure increase of OpenMP threads, dashed line) and the improved variant (mixed MPI decomposition and increase of OpenMP threads). The test case consists of $6912 \times 600 \times 256$ grid points in the x-, y-, and z-direction, respectively. For the baseline case the domain is split into 768 MPI sub-domains in the $x - y$-plane using one CPU per MPI process. The scaling is then done using parallelization of the z-direction up to 24 CPUs, giving a scaling from 768 to 18432 CPUs total. Very good results have been achieved showing a large increase in efficiency for high core numbers and thus enabling even higher parallelization while keeping the already-good base performance of the code, see [30]. The scaling using parallelization of the z-direction is now on par with scaling (by pure domain decomposition) in the x-y plane, see [9].

NEC Aurora. A NEC-SX-Aurora system was recently installed at the HLRS. The system consists of eight nodes, each with eight NEC vector engine cards (Type 10B), for a total of 512 vector cores. Although in-depth performance analysis and optimization is still underway, initial experience with the system is very good. The NS3D code was able to achieve up to 12% of the theoretical peak performance on large cases run on 256 cores. The major code change so far has been the replacement of a scalar random-number generator with one from the NEC Numeric Library Collection. The only drawback has been the amount of memory per vector engine, which in some cases has been a limiting factor for the size of the simulation.

Performance Optimization Workshop. The HLRS biannually hosts the "Optimization of Scaling and Node-Level Performance" workshop, where code developers jointly work with the cluster experts of the supercomputing center on their simulation software. The knowledge gained during these workshop has greatly helped in the

performance optimization of the revised NS3D code and its scaling aspects. Due to the time investment from both the people working on NS3D and the team of HLRS, see the acknowledgements, the code is well suited for current and next-generation HPC systems at HLRS.

4 DNS Results

In the present paper, the main focus is on the data compression algorithm applied to the DNS data. Hence, only the main flow characteristics of the DNS conducted are presented in the following to showcase the chosen set-up with its long domain. To this end, both the spatial evolution of the skin-friction coefficient $c_f = 2\overline{\tau}_w/(\overline{\rho}_w u_e^2)$ and the skin-friction Reynolds number are given in Fig. 3(a) and (b), respectively, both plotted versus Re_θ. The incompressible correlations are given by $c_f = 0.024 Re_\theta^{-1/4}$ [21] and $Re_\tau = 1.13 Re_\theta^{0.843}$. The compressible correlation for c_f is computed from the van-Driest2 transformation [27], for Re_τ by a simple shift of the incompressible reference. Incompressible reference data are taken from [20] and are depicted as red colored open symbols. Compressible reference data are taken from [18] for $M_\infty = 2.0$ and are in very good agreement with the present data; both results hence can be denoted to be of excellent quality. It is mentioned, however, that the compressible reference data have been computed in three different simulations, $872 \leq Re_\theta \leq 1242$, $2082 \leq Re_\theta \leq 2921$ and $4430 \leq Re_\theta \leq 6071$,

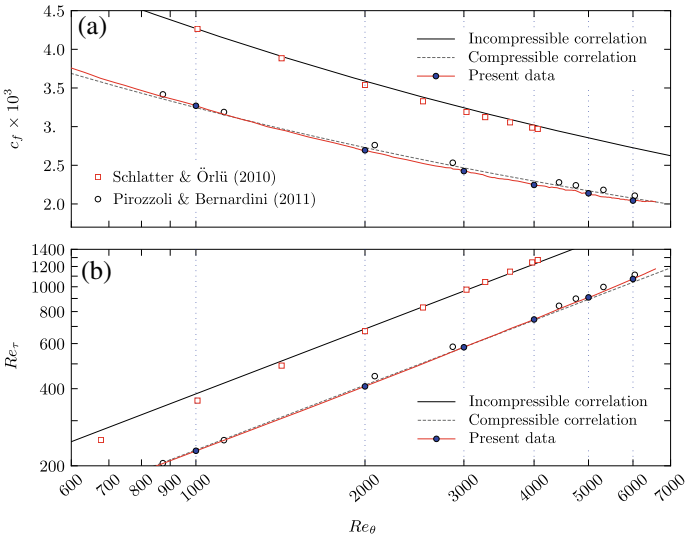


Fig. 3 **a** Skin-friction coefficient c_f and **b** skin-friction Reynolds number Re_τ as function of the momentum-thickness Reynolds number Re_θ

respectively. Each simulation is therefore again influenced by the turbulent inlet boundary condition which slightly affects the results, see [31] for details. On this background, the newly generated data gained by one long domain represent a clear improvement of the status quo, both in terms of the reliability of the data obtained and the maximum Reynolds number achieved. The use of only simulation domain additionally allows to both observe and analyze the spatial development of turbulent structures in a reliable manner.

It should be pointed out again, however, that the available storage capacities are not sufficient to carry out the proposed investigations on the large data that will be available to turbulence researchers. From this point of view, it is absolutely necessary to be equipped with a compression algorithm which, depending on the application, is easy to use and offers high compression ratios without dissipating vital flow field information. Otherwise, the potential of the increased computing capacity will not be usable and the intended investigations in future will still be limited to low Reynolds number ranges.

5 Data Compression Algorithm

The compression algorithm implemented in the BigWhoop library has been derived from the JPEG-2000 standard at the IAG/HLRS. It is designed around the discrete wavelet transform, facilitating random access as well as region-dependent distortion-control operations [24]. Furthermore, its embedded block coding algorithm offers an optimally truncated, quality- and resolution scalable code stream. Combined, these features make JPEG-2000 a standard that can be used to efficiently compress a wide variety of CFD problems without sacrificing crucial flow-field information.

The fundamental structure of the BigWhoop compressor is shown in Fig. 4. Since the underlying algorithm has been adapted from a standard defined for integer values, the first course of action is to transform the volumetric floating-point into an integer array. This is accomplished by using the floating-to-fixed-point transfrom described

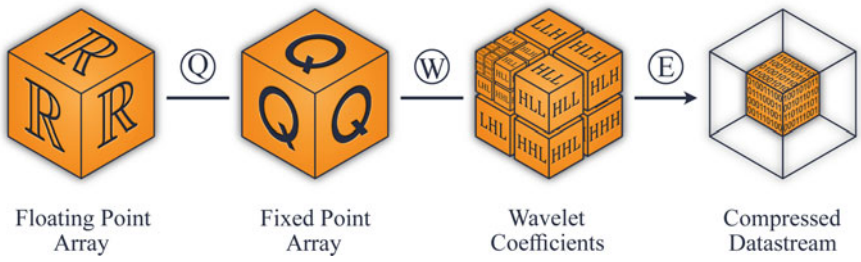


Fig. 4 Structure of the compression algorithm implemented in the BigWhoop library. Encircled letters indicate the floating-point to fixed-point transform (Q), discrete wavelet transform (W) and entropy encoding stage (E)

in *Fixed-Point Number Format Q*. The next step is to derive a time-frequency representation from the transformed data samples. Here, the discrete wavelet transform is applied to the volumetric integer field to decorrelate its inherent spatial frequency information (see *Discrete Wavelet Transform*). Finally, the wavelet coefficients are quantized and encoded using the Embedded Block Coding with Optimized Truncation (EBOCT) algorithm as described in *Entropy Encoding*. For a thorough discussion of the JPEG-2000 standard the reader is referred to the treatise by Taubman and Marcellin [24].

Fixed-Point Number Format Q. In numerical analysis, floating-point numbers are adopted to approximate the large dynamic range of real numbers used to describe fluid flow phenomena. The non-linearity of the mantissa, however, will introduce high-frequency information into the binary representation of numerical values. This, in turn, can degrade the performance of a compression algorithm [8, 26]. To bypass this problem, the number format Q is used to transform the data samples to a fixed-point representation with constant resolution [2, 15]. To this end, the dynamic range of each flow-field variable is first centered around zero. This is done to reduce the number of significant bits needed to represent the data samples and exploit the full range of the number format Q. Next, all values are normalized to the range $(-1, +1)$ and then multiplied by the number of fractional bits Q_m used in the fixed-point representation. In this context, the number of fractional bits should not exceed the width of the integer type used to store the fixed-point values. For our purpose, we use $Q_m = 62$ to prevent an integer overflow during the subsequent discrete wavelet transform (DWT) stage. Lastly, all samples are quantized and stored in a volumetric integer array.

Discrete Wavelet Transform. After transforming the volumetric floating-point into an integer array, the lifting-based, one-dimensional discrete wavelet transform is applied to the data set. In this context, the DWT is responsible for transforming the integer array into a time-frequency representation. This, in turn, concentrates the internal information into a small number of wavelet coefficients and extracts the location of high-frequency information that can be omitted during rate control.

The forward wavelet transform is best understood as a pair of low- and high-pass filters, commonly known as the analysis filter bank, followed by a down-sampling operation by a factor of two. This ensures a critically sampled signal that is split into a low- and high-frequency band (see Fig. 5). The low-pass filter attenuates high-frequency information, creating a blurred version of the original data set. This low-frequency band should represent a highly correlated signal which can be subjected to further wavelet decompositions, producing a so-called dyadic decomposition. The high-pass filter, on the other hand, preserves the high-frequency information that has been discarded in the low-frequency band. Preferably, the high-frequency band is sparsely occupied, resulting in a highly decorrelated signal [1, 4].

The discrete wavelet transform used for the current investigation is the 9-tab/7-tab real-to-real filter bank, commonly known as the 9/7-CDF-Wavelet. The 9/7-filter bank is split into two predictor (high-band) and two update (low-band) operations, followed by a dual normalization step:

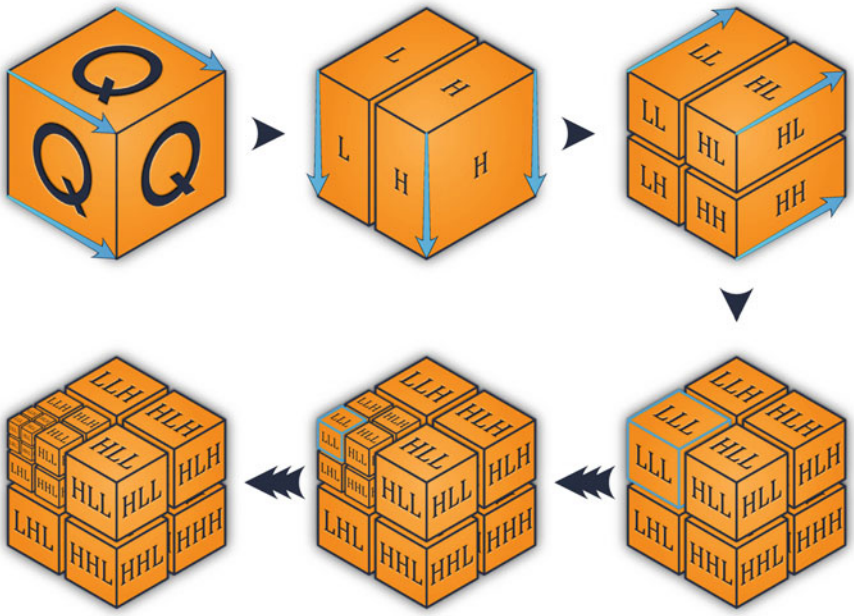


Fig. 5 Dyadic decomposition of a 3-dimensional numerical data set. The blue lines indicate direction of the one dimensional wavelet transform. The position of letters H and L in the sub-band label imply directionality of high- and low-frequency information

$$
\begin{aligned}
y(2n+1) &\leftarrow x(2n+1) + (\alpha \times |x(2n) + x(2n+2)|),\\
y(2n) &\leftarrow x(2n) + (\beta \times |y(2n-1) + y(2n+1)|),\\
y(2n+1) &\leftarrow y(2n+1) + (\gamma \times |y(2n) + y(2n+2)|),\\
y(2n) &\leftarrow y(2n) + (\delta \times |y(2n-1) + y(2n+1)|),\\
y(2n+1) &\leftarrow -K \times y(2n+1),\\
y(2n) &\leftarrow (1/K) \times y(2n).
\end{aligned}
\tag{1}
$$

In this context, $\alpha = -1.59$ and $\gamma = 0.88$ are the coefficients for the first and second predictor stage. The coefficients $\beta = -0.053$ and $\delta = 0.44$ define the first and second update stage. The dual normalization step is described by the coefficient K = 1.23 [19].

Since the discrete wavelet transform is a one-dimensional transformation defined for unbounded signals, we can extend the transformation stage to volumetric data sets by applying the analysis filter bank along each spatial dimension successively. For bounded data sets, the undefined samples outside of the finite-length signal need to be related to values inside the signal segment. For odd-length filter taps, this is achieved by applying a whole-point symmetric extent on the signal boundaries [24].

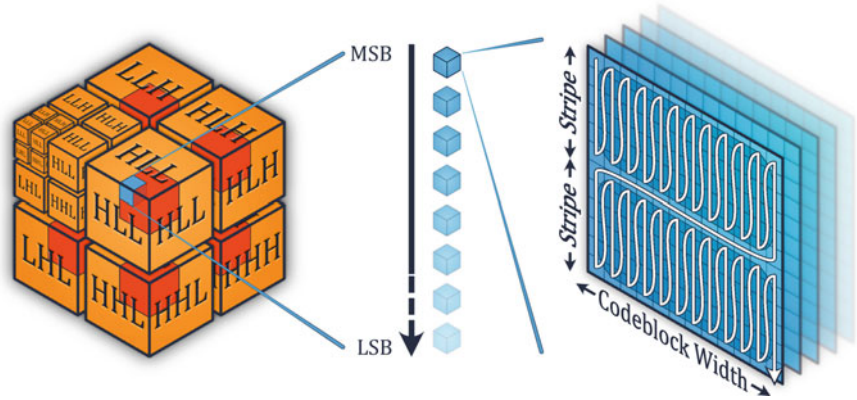


Fig. 6 Geometric operations performed during Embedded Block Coding with Optimized Truncation (EBCOT) stage. Red squares signal a precinct, blue squares a code block

Entropy Encoding. After the transformation stage, each wavelet sub-band is independently encoded using the EBCOT algorithm described in the JPEG 2000 standard. First, the wavelet coefficients are rounded down to the nearest integer. Next, each sub-band is subdivided into non-overlapping cubes (see Fig. 6). For every wavelet level, spatially related rectangles from the 15 high-bands form a precinct. Each precinct is further divided into $32 \times 32 \times 32$ code blocks, which form the input signal of the entropy encoding stage. These code blocks are then split into their bit-plane fields. The bit planes are scanned in a zigzag pattern, starting from the most significant bit plane (MSB) to the least significant bit plane (LSB). Each bit plane is encoded using three distinct coding passes:

The significant propagation pass will encode the current bit of a coefficient if one of its neighbors has become significant in a previous coding pass. To this end, the bit is first assigned to one of nine context labels based on the current sub-band as well as the number of its significant neighbors. The context information, alongside the current bit, is then delivered to the arithmetic MQ-Coder.

The refinement pass, on the other hand, will encode the bits for all coefficients that have become significant in a previous coding pass. Here, the context is assigned according to the significant state of the horizontal, vertical and diagonal direction. Similar to the significant propagation pass, the context label is then passed on to the arithmetic encoder to code the current bit.

The cleanup pass encodes all the bits that have not been encoded in the significant propagation and refinement pass. Run coding and context labels are used to encode the string of leftover bits.

During the coding operation, every bit of a bit plane is only encoded once in one of the three coding passes. The sign bit is encoded once a coefficient becomes significant for the first time.

In a post-processing operation, the encoded bit stream is subdivided into so-called quality layers. These quality layers represent optimized code-stream truncation points

that minimize the distortion for a specific bit rate. The encoded information for each code block is distributed across all layers. Rate control is handled by defining quality layer 0 to be rate-distortion optimized for the specified compression ratio. Each successive quality layer will monotonically improve the quality of the data-set reconstruction. Conversely, omission of subsequent quality layers from the code stream can be used to further reduce the size of a compressed data set. For this, however, multiple quality layers must already be defined during the compression.

6 Data Compression Analysis

In this section, the compression library is applied to a single time step of the previously described, unsteady flow field. To ensure optimal data reduction, the compression algorithm is applied to the entire computational domain as a whole. Decompression is then carried out using the full code stream to properly reconstruct the numerical data set. Thus, the distortion introduced by the lossy compression stage can be evaluated by calculating the difference between the original (u_o) and reconstructed (u_r) flow fields in the streamwise velocity component.

Results for the absolute $(u_o - u_r)/u_\infty$ and relative error $(u_o - u_r)/u_o$ are given in Fig. 9 and Fig. 10, respectively. Both plots have the same structure and allow for the assessment of the following influencing parameters:

(1) The performance of the BigWhoop library is analyzed by reconstructing the flow field from code streams that have been created using three different compression ratios—1 : 50, 1 : 100 and 1 : 200. The absolute and relative errors for the three reconstructions are evaluated and the results plotted in the subplot rows of Fig. 9 and 10, respectively.
(2) The influence of the local Reynolds number is analyzed using two sub-domains that are located near the flow field inlet ($Re_\theta = 1000$) and outlet ($Re_\theta = 6000$) (see Fig. 7). Results for the two subsets are depicted in the first ($Re_\theta = 1000$) and second ($Re_\theta = 6000$) subplot columns of Fig. 9 and 10. To enable an intuitive assessment of the size of the turbulent structures contained in both sub-domains, the corresponding temperature fields for the entire flow field, as well as the two sub-domains, are shown in Fig. 8.

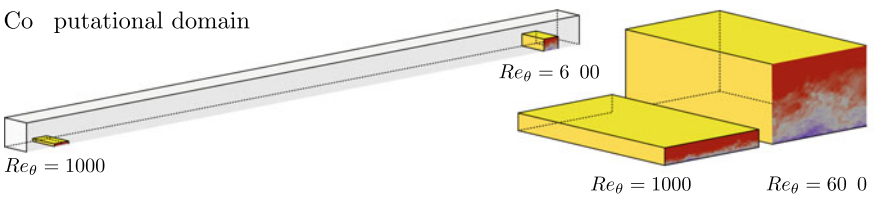


Fig. 7 Visualization of the computational domain and two close-ups of the sub-domains at $Re_\theta = 1000$ and 6000, which are used to evaluate the compression-code performance

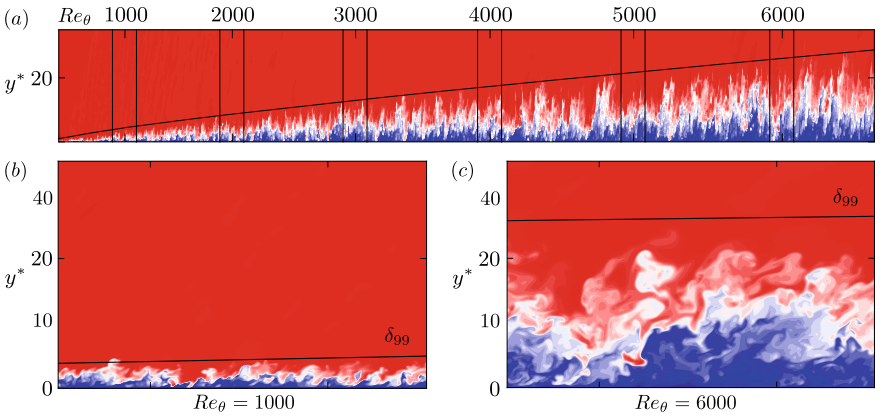


Fig. 8 Temperature distribution of the unsteady flow field for the complete length of the simulation domain **a** and subsets extracted at $Re_\theta = 1000$ **b** and $Re_\theta = 6000$ **c**. The temperature distribution is colored from red to blue from $T/T_\infty = 1$ in the far-field to $T/T_\infty = 1.714$ at the adiabatic wall

(3) The influence of subdividing the flow field into tiles before compression is evaluated by reconstructing the sub-domain at $Re_\theta = 6000$ from code streams that have been created using three different compression ratios—1 : 50, 1 : 100 and 1 : 200. The absolute and relative error for this case are evaluated and outlined in the third column of Figs. 9 and 10, respectively.

Values for the absolute and relative errors are plotted on the abscissa of Fig. 9 and 10. The ordinate represents the relative number each value can occur in its respective domain—the error occurence probability. To assess the evolution of the compression-induced distortion over the height of the turbulent flow, each domain is subdivided into wall-parallel slices. All slices have a height of 10 grid points in the wall-normal direction and are plotted individually as a colored line. Colors are chosen to vary from dark blue, corresponding to the slice directly at the wall, to dark red, representing the slice at the edge of the boundary layer. All results are averaged in the three spatial directions for every slice.

Evaluation. We will first discuss the absolute error $(u_o - u_r)/u_\infty$ shown in Fig. 9. For the sub-domain at $Re_\theta = 1000$ (subplot (a)–(c)), the maximum error is located in the direct vicinity of the wall (blue lines). The maximum values are observed to be 0.007%, 0.05% and 0.2% for the compression ratios of 1 : 50, 1 : 100 and 1 : 200, respectively. For larger wall distances, where the size of turbulent eddies increases, the absolute error decreases. A comparison between the results for $Re_\theta = 1000$ (subplot (a)–(c)) and $Re_\theta = 6000$ (subplot (d)–(f)) shows comparable results and therefore suggests that the errors introduced by the lossy compression are not subject to a large Reynolds number effect. In contrast, results for the reconstructed sub-domain (see Fig. 9 subplot (g)–(i)) exhibits a maximum error of 0.07, 0.2 and 0.5% for the three compression ratios—a significant increase when compared to the complete domain. This behaviour is attributed to the strong difference in information

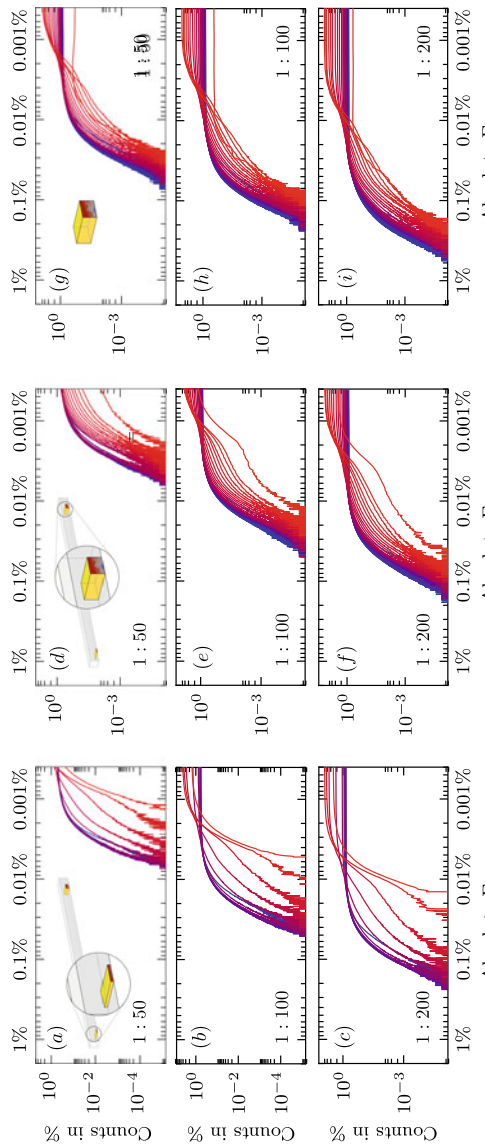


Fig. 9 Absolute error $(u_o - u_r)/u_\infty$ between original u_o and reconstructed velocity field u_r for compression ratios of 1 : 50, 1 : 100 and 1 : 200. 'Counts' on the ordinate axis denotes the occurrence probability of the error. For the left two columns, the complete simulation domain is compressed. Results are evaluated for a small sub-domain at $Re_\theta = 1000$ in (*a*)–(*c*) and at $Re_\theta = 6000$ in (*d*)–(*f*). For the right column, only the sub-domain at $Re_\theta = 6000$ is compressed. The error is computed for wall-parallel slices with a thickness of 10 respective numerical grid points; results at the wall are depicted in blue, near the edge of the boundary layer in red

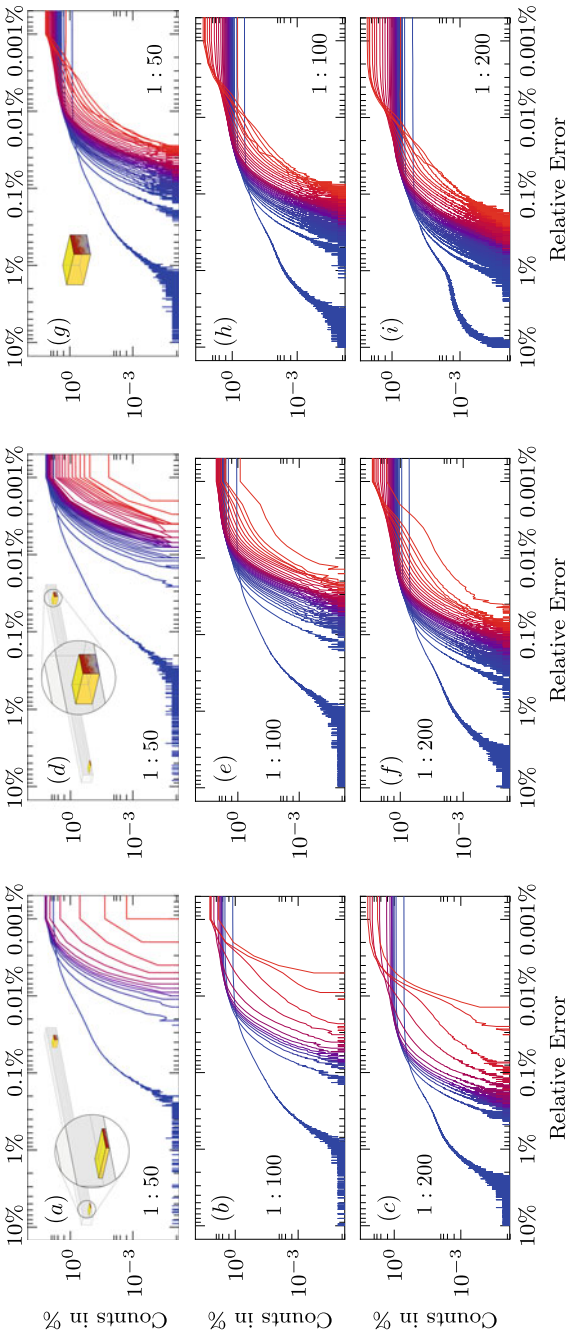


Fig. 10 Same as Fig. 9, but for the relative error $(u_o - u_r)/u_o$

density contained in the different domains. For the entire computational domain, the proportion of turbulent boundary layer (high information density) to non-turbulent far-field flow (low information density) is very low. The sub-domain at $Re_\theta = 6000$, in contrast, contains almost exclusively turbulent flow (high information density). Since the compression algorithm aims to minimize the local error according to the user specified average bit rate, the low information content of the far-field region lends itself to a far more effective compression than turbulent regions inside the boundary layer.

The relative error $(u_o - u_r)/u_o$ given in Fig. 10 follows a similar trend to the absolute error discussed in the previous paragraph. The no-slip boundary condition, however, forces the velocity field to zero at the wall. Since the relative error scales inversely with the local velocity u_o, we can observe very large values in wall adjacent data samples. Should we choose to ignore these large outliers, the maximum relative errors of the complete reconstruction were approximately 0.2%, 1.0% and 4% for the compression ratio of 1 : 50, 1 : 100 and 1 : 200 (see Fig. 10 subplot (a)–(f), respectively). If only the sub-domain at $Re_\theta = 6000$ is reconstructed, a maximum error of up to 10% is found in (g)–(i).

Assessment. To allow for an honest evaluation of the results shown and to estimate maximum achievable compression ratios for the purpose of turbulent boundary layers, some points have to be mentioned.

(1) All data shown are only evaluated for one single time step. For many post-processing purposes, which mostly use some kind of time averaging, the averaged error will be much smaller than the instantaneous one discussed so far.
(2) The maximum error detected is mainly located directly at the wall. For many post-processing purposes focusing on the spatial evolution of turbulent superstructures, which are mainly located in the outer part of the boundary layer, the error at the wall is irrelevant.
(3) At the same time, the incidence of erroneous data prints is rather low, typically less than 1%.

7 Conclusions and Outlook

Already today the discrepancy between the size of data output and the available storage capacity for very large simulations is becoming apparent. In order to fully exploit the continuously increasing computing capacity for DNS in turbulence research, data compression algorithms must be used which are able to compress the data output to an enormous degree. In this paper a performance analysis of the compression algorithm *BigWhoop* is applied to compressible-turbulent-boundary-layer data gained by DNS. The DNS data are computed for a Mach number of $M = 2.0$ and cover a Reynolds-number range of $Re_\theta = 300$ to 6600 and hence represent the longest simulated Reynolds-number range in terms of Re_θ available in the literature for the compressible regime. The numerical simulation consists of about 20.5 billion numer-

ical grid points, yielding a data size of about 1 TB for each time step including six flow-field variables.

To evaluate the compression potential of *BigWhoop* for this data set, data of one single time step of the unsteady flow field are compressed and decompressed for the whole computational domain. By calculating the difference between the original and the compressed-decompressed flow fields, u_o and u_r, respectively, the error made within the compression algorithm is computed for the streamwise velocity component. It is shown that DNS data compressed up to rates of $1:200$ yield a maximum absolute error $(u_o - u_r)/u_\infty$ of 0.2% in the near-wall region, which is still accurate enough to reliably perform flow-field investigations in the outer layer of the turbulent boundary layer, where the error is orders of magnitude smaller. Due to the no-slip boundary condition at the wall, forcing the velocity to decrease, the relative error $(u_o - u_r)/u_o$ is significantly larger in the close-wall region. Hence, for flow-field investigations depending on wall properties like the skin-friction factor, for instance, a compression rate of $1:100$ yields a maximum relative error of about 1%. Even if this error appears unacceptably large, it is mentioned that the occurance of the maximum error is extremely scarce and the time-averaged error is orders of magnitude smaller than the maximum unsteady local error. An investigation of the compression rate on the temporally averaged error is currently under investigation and shows that time-averaged errors caused by compression rates of $1:100$ are still insignificant. It should be noted, however, that the performance of the compression algorithm largely depends on the domain size chosen and the level of turbulent information contained in the domain compressed. For small domain sizes containing a lot of small-scale turbulent fluctuations, the compression potential is lower if comparable errors are targeted.

Outlook. To fully exploit the compression potential of *BigWhoop*, the present study has shown that an additional functionality appears to be useful, which allows the compression rate to be set separately for individual regions of interest. For data of turbulent boundary layers this would mean that the compression rate should be lower at the wall and successively increased with increasing wall distance. A corresponding option is already implemented in a revised version of *BigWhoop*.

Acknowledgements The financial support by the Deutsche Forschungsgemeinschaft, DFG, under reference number RI680/38-1, WE6803/1-1, SFB/TRR 40-A4, and the provision of computational resources on Cray XC40 by the Federal High Performance Computing Center Stuttgart (HLRS) under grant GCS Lamt (LAMTUR), ID 44026, are gratefully acknowledged.

References

1. T. Acharya, P.S. Tsai, *JPEG2000 Standard for Image Compression: Concepts, Algorithms and VLSI Architectures* (Wiley, New Jersey, 2005)
2. ARM. N.N. ARM Developer Suite, AXD and armsd Debuggers Guide (Cambridge, 2001)
3. A. Babucke, Direct numerical simulation of noise-generation mechanisms in the mixing layer of a jet. Ph.D. thesis, University of Stuttgart, 2009

4. C. Christopoulos, A. Skodras, T. Ebrahimi, The JPEG2000 still image coding system: an overview. IEEE Trans. Consum. Electron. **46**(4), 1103–1127 (2000)
5. T. Colonius, S.K. Lele, P. Moin, Boundary conditions for direct computation of aerodynamic sound generation. AIAA J. **31**(9), 1574–1582 (1993)
6. P.C. Dörr, Z. Guo, J.M.F. Peter, M.J. Kloker, Control of traveling crossflow vortices using plasma actuators, in *High Performance Computing in Science and Engineering '17*. ed. by W.E. Nagel, D.B. Kröner, M.M. Resch (Springer, Cham, 2017), pp. 255–267. https://doi.org/10.1007/978-3-319-68394-2_15
7. G. Eitel-Amor, R. Örlü, P. Schlatter, Simulation and validation of a spatially evolving turbulent boundary layer up to reθ= 8300. Int. J. Heat Fluid Flow **47**, 57–69 (2014)
8. M.N. Gamito, M.S. Dias, Lossless coding of floating point data with JPEG 2000 part 10, in *Applications of Digital Image Processing XXVII, International Society for Optics and Photonics*, vol. 5558 (2004), pp. 276–287
9. M Keller, M.J. Kloker, DNS of effusion cooling in a supersonic boundary-layer flow: influence of turbulence, in *44th AIAA Thermophysics Conference* (2013). https://doi.org/10.2514/6.2013-2897. (aIAA-2013-2897)
10. M.A. Keller, M.J. Kloker, DNS of effusion cooling in a supersonic boundary-layer flow: influence of turbulence. AIAA-2013-2897 or AIAA paper 2013-2897 (2013)
11. M.A. Keller, M.J. Kloker, Effusion cooling and flow tripping in laminar supersonic boundary-layer flow. AIAA J. **53**(4), 902–919 (2014)
12. M.A. Keller, M.J. Kloker, Direct numerical simulation of foreign-gas film cooling in supersonic boundary-layer flow. AIAA J. **55**(1), 99–111 (2016)
13. M. Klein, A. Sadiki, J. Janicka, A digital filter based generation of inflow data for spatially developing direct numerical or large eddy simulations. J. Comput. Phys. **186**, 652–665 (2003)
14. H.B.E. Kurz, M.J. Kloker, Receptivity of a swept-wing boundary layer to micron-sized discrete roughness elements. J. Fluid Mech. **755**, 62–82 (2014)
15. P. Lindstrom, M. Isenburg, Fast and efficient compression of floating-point data. IEEE Trans. Visual Comput. Graph. **12**(5), 1245–1250 (2006)
16. J. Linn, M.J. Kloker, Numerical investigations of film cooling, in *RESPACE—Key Technologies for Resuable Space Systems. NNFM*. ed by A. Gülhan, vol. 98 (2008), pp. 151–169
17. J. Linn, M.J. Kloker, Effects of wall-temperature conditions on effusion cooling in a supersonic boundary layer. AIAA J. **49**(2), 299–307 (2011)
18. S. Pirozzoli, M. Bernardini, F. Grasso, Direct numerical simulation of transonic shock/boundary layer interaction under conditions of incipient separation. J. Fluid Mech. **657**, 361–393 (2010)
19. M. Rabbani, R. Joshi, An overview of the JPEG2000 still image compression standard. Signal Process.: Image Commun. **17**(1), 3–48 (2002)
20. P. Schlatter, R. Örlü, Assessment of direct numerical simulation data of turbulent boundary layers. J. Fluid Mech. **659**, 116–126 (2010)
21. A.J. Smits, N. Matheson, P.N. Joubert, Low-Reynolds-number turbulent boundary layers in zero and favorable pressure gradients. J. Ship Res. **27**(3), 147–157 (1983)
22. A.J. Smits, B.J. McKeon, I. Marusic, High-Reynolds number wall turbulence. Ann. Rev. Fluid Mech. **43**, 353–375 (2011)
23. P.R. Spalart, Direct simulation of a turbulent boundary layer up to Re_θ= 1410. J. Fluid Mech. **187**, 61–98 (1988)
24. D. Taubman, M. Marcellin, *JPEG2000: Image Compression Fundamentals, Standards and Practice* (Springer, New York City, 2002)
25. T. Trader, STREAM benchmark author McCalpin traces system balance trends (2016). https://www.hpcwire.com/2016/11/07/mccalpin-traces-hpc-system-balance-trends/
26. B. Usevitch, JPEG2000 compatible lossless coding of floating-point data. EURASIP J. Image Video Process. **2007**(1), 085,385 (2007)
27. E.R. Van Driest, On turbulent flow near a wall. J. Aeronaut. Sci. **23**(11), 1007–1011 (1956)
28. M.R. Visbal, D.V. Gaitonde, On the use of higher-order finite-difference schemes on curvilinear and deforming meshes. J. Comput. Phys. **181**, 155–185 (2002)

29. C. Wenzel, DNS of compressible turbulent boundary layers: pressure-gradient influence and self-similarity. Disseration, University of Stuttgart, 2019
30. C. Wenzel, J.M.F. Peter, B. Selent, M.B. Weinschenk, U. Rist, M.J. Kloker, DNS of compressible turbulent boundary layers with adverse pressure gradients, in *High Performance Computing in Science and Engineering 2017*. ed. by W.E. Nagel, D.B. Kröner, M.M. Resch (Springer, Cham, 2018), pp. 229–242. https://doi.org/10.1007/978-3-030-13325-2_14
31. C. Wenzel, B. Selent, M.J. Kloker, U. Rist, DNS of compressible turbulent boundary layers and assessment of data/scaling-law quality. J. Fluid Mech. **842**, 428–468 (2018)

Numerical Analysis of Ethanol and 2-Butanone Direct Injection in an Internal Combustion Engine

Tim Wegmann, Sven Berger, Matthias Meinke, and Wolfgang Schröder

Abstract The mixture formation of air and the gasoline surrogates Ethanol and 2-Butanone is numerically investigated using large-eddy simulations of the direct-injection into an internal combustion engine. The multiphysics solver m-AIA (multiphysics- Aerodynamisches Institut Aachen) is used, which is based on a finite-volume method with a fully coupled Lagrangian particle tracking model for fuel spray applications formulated for hierarchical cartesian grids. The validation of the spray model is performed by comparison with pressure chamber measurements, where a good agreement for the spray penetration is obtained. PIV Measurements of the in-cylinder flow-field are used to validate the flow field simulation results in the internal combustion engine by comparing cycle averaged data for the flow field. The mixture formation for the two biofuels is investigated under engine conditions. The higher evaporation rate of 2-Butanone leads to significant differences in the spray distribution. The lower liquid penetration of the 2-Butanone spray together with unfavorable three-dimensional flow-field effects lead to a non-optimal mixture formation for 2-Butanone at the start of ignition.

1 Introduction

Future use of internal combustion (IC) engines requires the usage of crude oil replacements to ensure CO_2 neutral transportation [8]. Biofuels as energy carriers are such a replacement, but pose new challenges due to their different thermo-physical properties such as viscosity, heat capacity among many other influencing factors. The

T. Wegmann (✉) · S. Berger · M. Meinke · W. Schröder
Institute of Aerodynamics, RWTH Aachen University, Wüllnerstraße 5a, 52062 Aachen, Germany
e-mail: t.wegmann@aia.rwth-aachen.de

W. Schröder
JARA Center for Simulation and Data Science, RWTH Aachen University, Seffenter Weg 23, 52074 Aachen, Germany

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_20

geometry and process parameters of IC engines have to be optimized for the specific biofuel to simultaneously obtain high engine efficiency and low emissions [5, 20].

Such engine optimizations require numerical models, which are capable of accurately predicting the engine performance for e.g. different cylinder head and piston geometries, valve, and injection parameters. To resolve all relevant scales of the mixture formation turbulence scale resolving methods such as large-eddy simulations (LES) are required. Consequently, high-performance computing hardware is required for the high mesh resolution. The time varying geometry in the engine and the moving valves need adaptive mesh refinement consequently dynamic load balancing is required to achieve a high computational efficiency.

The multiphysics solver [9] developed at the Institute of Aerodynamics RWTH Aachen University, offers such Cartesian mesh based methods with cut-cell boundary formulations. Unstructured Cartesian meshes are used, since they allow automatic mesh generation and a more efficient implementation of solution adaptive meshes compared to body-fitted mesh approaches. Dynamic load balancing is achieved by using a space-filling curve on the base level of the hierarchical octree mesh for the domain decompositioning.

In this paper, simulations of the spray formation are performed for the injection of Ethanol and 2-Butanone, part of which have been experimentally studied by Braun et al. [4]. The differences in mixing formation of the two biofuels will be discussed at the time of ignition by assessing the concentration and fuel distribution functions. Note that this contribution is based on the conference paper published in [3].

The structure of the paper is as follows, first in Sect. 2 the models of the finite-volume, Lagrangian particle tracking and injection are described. Next, Sect. 3 provides details of the applied algorithms. Subsequently, in Sect. 4 the setup used by engine and spray simulations is described. In Sect. 5 the required computing resourced are mentioned. Finally, the results are presented in Sect. 6 and concluding remarks are given in Sect. 7.

2 Mathematical Model

In the following the mathematical model for the LES of the continuous phase is given in Sect. 2.1. Subsequently, the spray models for the disperse phase, which are two way coupled with the flow field in the IC engine are summarized in Sect. 2.2.

2.1 Continuous Phase

The motion of a compressible, viscous fluid is governed by the conservation of mass, momentum, and energy (i.e. the Navier-Stokes equations). For dual-species flow, the advection-diffusion equation of the species concentration Y needs to be included. The set of equations can be formulated in the arbitrary Lagrangian-Eulerian (ALE)

formulation as

$$\frac{d}{dt}\int_{V(t)} \mathbf{Q}\, dV + \oint_{\partial V(t)} \left(\overline{\mathbf{H}}_{\text{inv}} + \overline{\mathbf{H}}_{\text{vis}}\right) \cdot \mathbf{n}\, dA = \mathbf{S}, \tag{1}$$

where $V(t) \subset \Omega(t)$ is a moving control volume bounded by the surface $\partial V(t)$ with the outward pointing normal vector $\mathbf{n}$. The vector of conservative variables $\mathbf{Q} = [\rho, \rho\mathbf{u}, \rho E, \rho Y]^T$ contains the gas phase density ρ, velocity vector $\mathbf{u}$, total specific energy per unit mass E and fuel density ρY. The source term vector $\mathbf{S} = [S_m, \boldsymbol{S_i}, S_e, S_y]^T$ is used to couple the disperse phase and the gas phase. The inviscid $\overline{\mathbf{H}}_{\text{inv}}$ and viscous flux tensors $\overline{\mathbf{H}}_{\text{vis}}$ are given by

$$\overline{\mathbf{H}}_{\text{inv}} + \overline{\mathbf{H}}_{\text{vis}} = \begin{pmatrix} \rho(\mathbf{u} - \mathbf{u}_{\partial V}) \\ \rho\mathbf{u}(\mathbf{u} - \mathbf{u}_{\partial V}) + p\bar{\mathbf{I}} \\ \rho E(\mathbf{u} - \mathbf{u}_{\partial V}) + p\mathbf{u} \\ \rho Y(\mathbf{u} - \mathbf{u}_{\partial V}) \end{pmatrix} + \frac{1}{Re_0} \begin{pmatrix} 0 \\ \overline{\tau} \\ \overline{\tau}\mathbf{u} + \mathbf{q} \\ \frac{-\rho D \nabla Y}{Sc_0} \end{pmatrix}, \tag{2}$$

with the pressure p, the velocity of the control volume surface $\mathbf{u}_{\partial V}$, and the unit tensor $\bar{\mathbf{I}}$. All equations are non-dimensionalized by the stagnation state properties. The resulting Reynolds number given by $Re_0 = \underline{\rho_0}\,\underline{a_0}\,\underline{L}/\underline{\mu_0}$, using the speed of sound at rest $\underline{a_0} = \sqrt{\gamma\, \underline{p_0}/\underline{\rho_0}}$ with the heat capacity ratio of air $\gamma = 1.4$ and the characteristic length $\underline{L}$.

The Schmidt number in the viscous flux tensor for the passive scalar species is $Sc_0 = \underline{\mu_0}/(\underline{\rho_0}\underline{D_0}) = 1.0$ and the diffusion coefficient $D = D(t) = \mu(T)/\rho$. Since the amount of fuel in IC engines is typically low, the fuel concentration Y is assumed to have no impact on the gas properties.

Assuming a Newtonian fluid with zero bulk viscosity the stress tensor $\overline{\boldsymbol{\tau}}$ is expressed by

$$\overline{\boldsymbol{\tau}} = \frac{2}{3}\mu\,(\nabla \cdot \mathbf{u})\,\bar{\mathbf{I}} - \mu\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right) \;,\; \text{with} \quad \mu(T) = T^{3/2}\frac{1 + \underline{S}/\underline{T_0}}{T + \underline{S}/\underline{T_0}} \tag{3}$$

in which the dynamic viscosity μ at temperature T is determined by Sutherland's Law, where constant $\underline{S} = 111K$ is used for air at moderate temperatures. The heat flux based on Fourier's law with constant Prandtl number for air of $Pr_0 = 0.72$ reads

$$\mathbf{q} = -\frac{\mu}{Pr_0(\gamma - 1)}\nabla T. \tag{4}$$

The equations are closed by the ideal gas law in non-dimensional form $\gamma p = \rho T$.

2.2 Spray Model

The liquid phase is modelled by discrete parcels tracked by a Lagrangian model. To reduce the computational effort of Lagrangian particle tracking (LPT) methods, a number of fuel droplets/particles N_d is combined into a parcel which is tracked as a single Lagrange particle.

The equation of motion for parcel p with velocity $\mathbf{u}_p$ reads

$$\frac{d\mathbf{u}_p}{dt} = (1 - \frac{\rho}{\rho_p}) \cdot \mathbf{g} + \frac{C_D}{\tau_p} \frac{Re_p}{24} (\mathbf{u} - \mathbf{u}_p) \tag{5}$$

with the particle Reynolds number $Re_p = \frac{\rho ||\mathbf{u}-\mathbf{u}_p|| d_p}{\mu}$ and the particle relaxation time

$$\tau_p = \frac{\rho_p d_p^2}{18\mu}. \tag{6}$$

The drag coefficient C_D is given by [15] as

$$C_D = \begin{cases} \frac{24}{Re_p} & \text{for } Re_p \leq 0.1 \\ \frac{24}{Re_p}(1 + \frac{1}{6} Re_p^{\frac{2}{3}}) & \text{for } Re_p \leq 1000 \\ 0.424 & \text{for } Re_p > 1000 \end{cases} \tag{7}$$

with the position of the parcel determined by $\frac{d\mathbf{x_p}}{dt} = \mathbf{u}_p$. The source term $\mathbf{S}_i$ for the momentum exchange between the disperse and gas phase is given by

$$\mathbf{S}_i = N_d \cdot m_p \frac{d\mathbf{u}_p}{dt} \tag{8}$$

with the parcel mass m_p. The energy source term S_e of a fuel parcel is given by

$$S_e = 0.5\mathbf{S}_i \cdot \mathbf{u}_p + N_d \cdot c_{p_p} (m_p \cdot dT_p + dm_p \cdot T_p) \tag{9}$$

with the heat capacity of the fuel c_{p_p} and the parcel temperature T_p.

The mass source term S_m as well as the species source term S_Y is gained from the evaporation model of Belan and Harstad [1] which requires the solution the two additional differential equations for the evaporated parcel mass loss $\frac{dm_p}{dt}$ and parcel temperature $\frac{dT_p}{dt}$. In this version non-equilibrium effects are taken into account by introducing a correction term involving the Langmuir-Knudsen law. The implementation is based on the infinite liquid conduction version as proposed by Miller and Belan [12].

The primary breakup is based on a model proposed in [19] where the initial droplet Sauter mean diameter is calculated by mass conservation. A Rosin-Rammler distri-

bution is than used to get a larger variety of initial droplet sizes. Secondary breakup is model by the Kelvin-Helmholtz Rayleigh-Taylor (KHRT) model as described in [16].

3 Numerical Methods

A finite-volume method formulated for hierarchical Cartesian grids is used for the solution of the conservation equation Eq. (1) and a semi-Lagrange level-set solver is used to predict the evolution of the embedded moving boundaries of the piston and valves [6]. Due to different grid spacing requirements each solver uses an individual subset of the joint hierarchical grid, which is locally refined and adapted independently during the solution procedure. The Lagrangian particle solver uses the same decomposition as the finite-volume solver.

3.1 Finite Volume Solver

The turbulent flow field is predicted by an LES using a cell-centered finite-volume method. In the LES formulation only the larger turbulent structures are resolved and a monotone integrated LES (MILES) approach, in which the truncation of the scheme acts as the subgrid scale model, is used in this application to model turbulent scales smaller than the grid filter.

A low-dissipation variant of the advection upstream splitting method (AUSM) suitable for LES as reported in [11] is used to approximate the inviscid fluxes in the Navier-Stokes equations. The viscous flux tensor is approximated through a central-difference scheme, where gradients at the cell-surface centers are computed using the re-centering approach by Berger et al. [2]. Cell-centered gradients are computed by a weighted least-squared reconstruction scheme such that a second-order accurate scheme in space is obtained. Second-order accurate time integration is gained by an explicit five-stage Runge-Kutta scheme optimized for moving boundary problems and stability optimized Runge-Kutta coefficients as proposed by Schneiders et al. [17] are used. The time step Δt is computed via the Courant-Friedrichs-Lewy (CFL) constraint under consideration of the injection velocity $\mathbf{u}_{inj}$.

The moving boundaries are represented by a conservative sharp multiple-cut-cell and split-cell method as described in [6, 18]. A multiple-ghost cell approach allows the prescription of individual boundary conditions if a cut-cell is intersected by different surfaces. A combined interpolation and flux-redistribution technique is used to obtain a conservative and stable numerical method for small cells. Further details on the overall numerical method can be found e.g. in [6, 14, 17].

3.2 Level-Set Solver

The location of the moving surfaces is predicted by a multiple level-set solver, where each individual surface element can be represented as the zero contour of the i-th signed-distance function $\phi_i(\mathbf{x}, t)$. Different level-set functions are used individually for the various parts of the engine, e.g., cylinder head, piston and valves. The overall surface $\Gamma(t)$ is constructed from all $\Gamma_i(t)$ using the combined level-set $\phi_0(\mathbf{x}, t)$ as described in [7]. The distance from each interface can be easily evaluated by the absolute value of the level-set value $\phi_i(\mathbf{x}, t)$, which can be used to determine the gap width between interfaces approaching each other, i.e., when the valves are close to the valve seat. If the gap width falls below a specified threshold the combined $\phi_0(\mathbf{x}, t)$ level-set field in the vicinity of the gap can be modified to an artificially closed gap [7]. Since only the combined level-set is changed, no surface location information is lost through the modification.

For the translational motion of the piston and valves a semi-Lagrange solver [7] is used to determine the level-set function. In this approach the error of the geometric representation of the surface is reduced to the constant interpolation error between two reference locations.

3.3 Lagrangian Particle Solver

An implicit Euler method is used to solve Eqs. (5) and the differential equations for the evaporative mass loss $\frac{dm_p}{dt}$ and parcel temperature $\frac{dT_p}{dt}$ as mentioned in Sect. 2.2. The weights $w_i = \frac{d_i}{\sum_i d_i}$ with the distance d_i to the parcel over one layer of surrounding Cartesian cells are used to couple the particle solver with the flow solver, i.e., conservative flow variables and source term distribution.

Parcels are merged, if a single cell contains more than 250 parcels. The parcels closest to each other are merged. The new number of droplets and the particle diameter of the merged parcel are determined such that the Sauter mean diameter is preserved. Wall collisions are considered at present as hard-sphere collisions.

Since a large number of fuel droplet parcels has to be tracked for an accurate prediction of the fuel spray and due to the localized nature of the parcels in the spray plume, the computational effort changes locally during the spray injection, which is taken into account in the dynamic load balancing method.

4 Setup

All injection simulations are performed with a model for an outward opening hollow-cone injector, as used in the experiments published by Braun et al. [4]. The injector features a 4 mm injector nozzle diameter, while the design cone angle is 94° with a

maximum needle lift of 35 µm [10]. Parcels are generated on a 4 mm diameter circle and a Rosin-Rammler distribution for the initial droplet Sauter mean diameter (SMD) as primary-breakup model is applied. Fuel properties are obtained from [1, 4, 12] and *Spray G3* condition for early injection as provided by the Engine Combustion Network is used.

The injection mass flow rate is calculated based on the injected mass in [21] by assuming a linear mass flow rate change during injector opening and closing. The required opening- and closing time of the injector is determined from [10, 21]. The nozzle discharge coefficient $C_{nozzle} = 0.69$ is used.

The necessary fuel mass is determined for stoichiometric mixture conditions (i.e.$\lambda = 1$) of the air mass enclosed in the engine at ignition. Since the 2-Butanone air/fuel mass ratio is larger than that for Ethanol, the amount of injected fuel and the injection duration for Ethanol is larger compared to 2-Butanone.

The geometry of the direct injection optical gasoline research engine *TINA* of the Institute of Aerodynamics of RWTH Aachen University is used in all engine simulations. This 4-valve test-engine has been intensively used for experimental measurements, e.g. in [3, 4]. Valve lift and opening timings follow the standard intake valve timings. Simulations are conducted for an engine speed of 1500 rpm, bore of 0.075 m, stroke of 0.0825 m and an resulting compression ratio of 9.1:1. A solution and boundary refined mesh consisting of a uniform level and two higher refinement levels is used. Additionally, the parcel location is taken into account to refine the mesh to the smallest cell length of $\Delta x / D \approx 0.0028$ or 0.22 mm.

5 Computing Resources

The total number of level-set and finite-volume (FV) mesh cells changes strongly during the simulation cycle as the adaptive mesh-refinement changes the grid resolution based on moving piston and parcel location. The number of FV-cells for the engine setup without and with Ethanol injection for instance is shown in Fig. 1. The resulting average cell count of 36 million cells for the setup without injection is dis-

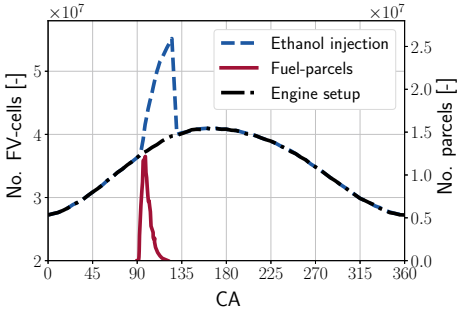


Fig. 1 Cell- and parcel numbers for the Ethanol engine injection simulation and the engine simulation without injection.

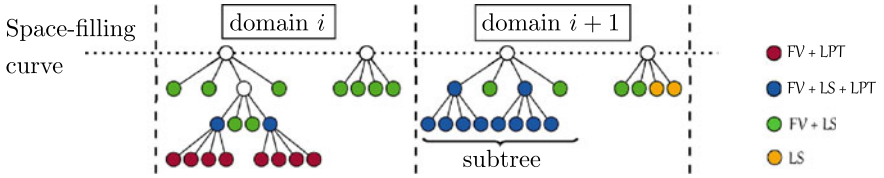


Fig. 2 Domain decomposition of a hierachical quadtree multi-physics grid with cell used finite volume (FV) discretization, lagrangian particle (LPT) coupling, level-set (LS) solver or a combination.

tributed to 50 compute nodes of the high-performance platform Hazelhen. As each node consists of two Intel® Xeon® E3-2680 v3 CPUs a total number of 1200 cores were used. Approx. 120 h are required for a complete cycle simulation.

Figure 1 also indicates the local and time dependent FV-cell distribution variation, which leads a to significant dynamic imbalance during the computation. Additionally, the differing grid resolution for the level-set and lagrangian particle solver and their differing computational costs per cell increase the imbalance. To achieve a high parallel efficiency dynamic load balancing is utilized based on a redistribution of the parallel subdomains [13]. Using a space-filling curve to partition the joint hierarchical Cartesian grid ensures that all solvers are coupled efficiently, i.e., no additional communication between solvers is necessary. An exemplary domain decomposition of a quadtree grid is depicted in Fig. 2.

After each mesh adaptation step, the load timers of the different partitions are gathered for 5 time-steps and used in the subsequent balancing step. As already shown in [13] this dynamic load balancing approach can be used to improve the scaling capabilities of the multi-physics solver.

6 Results and Discussion

First, spray simulations for the two biofuels in a pressure chamber setup are validated against experimental results in Sect. 6.1. Secondly, in-cylinder flow-field simulations are compared to experimental PIV measurements in the tumble-plane in Sect. 6.2. Finally, results from the biofuel injection in the engine are discussed in Sect. 6.3.

6.1 Pressure Chamber

For the validation of the spray model, a setup for a pressure chamber with the size $10 \times 10 \times 10\,\text{cm}^3$ is simulated and the results are compared to reference data taken from [4]. Simulation results will be discussed for 10 MPa injection pressure, an ambient temperature of 333 K and a fuel temperature of 293.15 K.

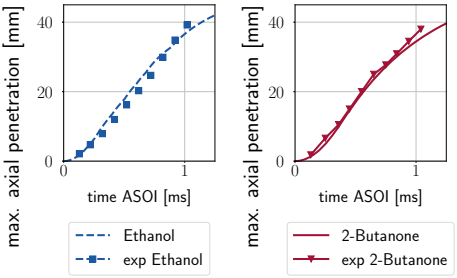


Fig. 3 Maximum axial penetration of the liquid spray as a function of time after start of injection (ASOI) compared to experimental results from [4].

A comparison of the axial penetration in Fig. 3 between experimental and simulation results shows a reasonably good agreement. Additional simulations with larger spatial resolution have shown an even better agreement with the experimental results between 0.4 ms and 0.8 ms after start of injection (ASOI). However, the high computational cost generated by the larger mesh size for the engine volume does not permit a higher spatial resolution for the spray.

A comparison of the evaporated fuel mass between the two fuels shows a higher evaporation rate for 2-Butanone which is in agreement with the results in [4]. At 1.25 ms ASOI, the amount of evaporated 2-Butanone is almost twice as high as of Ethanol. This consequence of the higher specific heat of vaporization of Ethanol can also be seen in the local concentrations in Fig. 4. Ethanol exhibits consistently about half the concentration value compared to 2-Butanone.

The flow and spray formation structures for the two fuels at constant pressure flow conditions show a strict axis symmetry and no significant difference besides the higher 2-Butanone concentration can be noted. Measurements show additional injector dependent effects resulting in slight variations of the liquid spray edge.

6.2 In-Cylinder Flow

The fv-solver for the continuous phase has already been shown to accurately capture the flow features in an axisymmetric single-valve engine [18]. In the following,

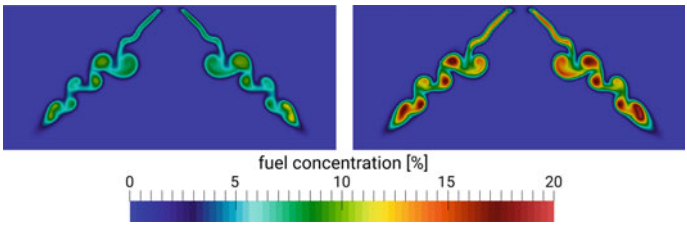


Fig. 4 Local fuel concentration at the end of injection for Ethanol (left) and 2-Butanone (right).

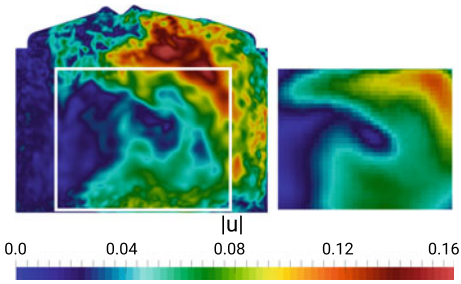


Fig. 5 Contours of the magnitude of the ensemble averaged velocity in the tumble plane obtained by the simulation (left) and experiment [3] (right) at 90° crank angles (CA). The experimental field-of-vision excludes the pent-roof and liner geometry and is denoted by the white lines in the simulation results of the full engine slice.

simulation results without spray injection and adiabatic walls are compared with PIV measurements in the tumble plane taken from [3].

The turbulent flow field results depend on the initial condition and thereafter vary from cycle to cycle, thus a meaningful comparison of the velocity field requires ensemble averaging. Therefore, five consecutive engine cycles were simulated and the results of the last four cycles were ensemble averaged. In the experiment, measurements of 125 engine cycles were used. Due to the smaller number of cycles in the simulation, only a qualitative agreement between measured and computed velocity field can be expected.

Figure 5 shows the non-dimensional magnitude of the velocity in the engine tumble plane. In the experiment the optical access is restricted to the region within the white rectangle depicted in the simulation results. Due to the small number of cycles the simulation results are not as smooth as the experimental data and larger velocity fluctuations are still visible. Yet a good qualitative agreement is visible. The velocity magnitude and also the locations of low and high velocity magnitude regions agree well with the experimental data.

The measurement plane and cycle averaged kinetic energy $E = \sum_{n=1}^{I} \frac{1}{2I} \left(\overline{u_n}^2 + \overline{v_n}^2 + \overline{w_n}^2\right)$ is compared to the experimental data in Fig. 6. Results are presented for a 3- and 4-cycle ensemble average. Due to larger velocity fluctuations in the simulation results a slightly higher averaged kinetic energy level compared to the experimental results is visible. This is confirmed by comparing the 3- and 4-cycle ensemble average, where an energy reduction for the larger number of cycles can be seen especially for crank angles with high turbulent kinetic energy i.e. CA below 90°.

In conclusion, the results from the engine simulation show a satisfactory agreement with the measurement data and further engine cycles are necessary for an even better agreement.

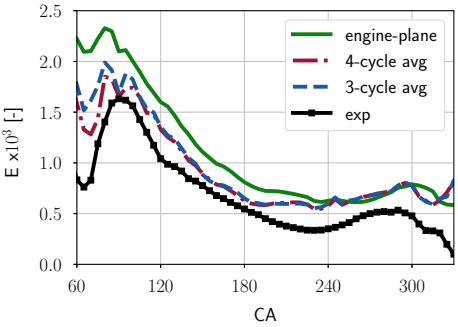


Fig. 6 Temporal evolution of the dimensionless ensemble- and plane-averaged kinetic energies E. Simulation results are either averaged in the measurement window of the tumble plane or in case of the engine-plane data the entire geometry slice including pent-roof and liner boundary. Experimental results are taken from [3].

6.3 Engine Injection

Next, the simulation results for the spray formation in the engine and the resulting mixing of the two biofuels will be discussed.

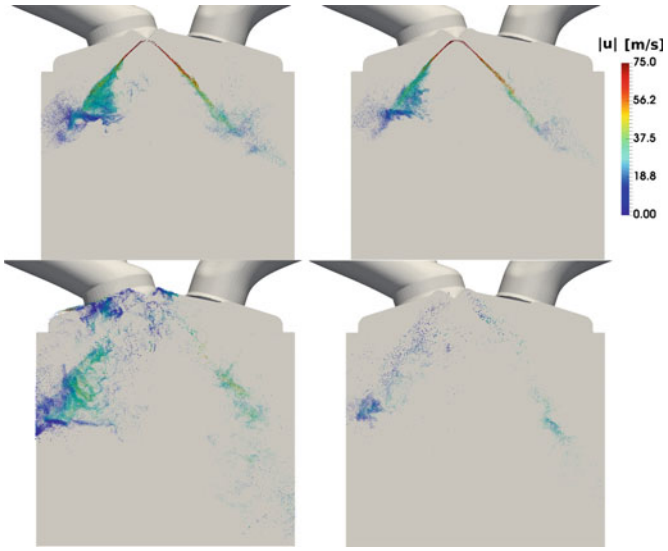


Fig. 7 Fuel parcels in the tumble plane for Ethanol (left) and 2-Butanone (right) at 100 CA (top) and 110 CA (bottom). Parcels are scaled to mass equivalent size while the color indicates the magnitude of the instantaneous velocity [m/s].

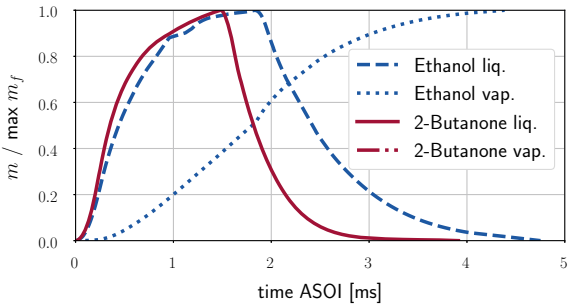


Fig. 8 Liquid and vapor mass of the fuel normalized by the respective maximum fuel mass.

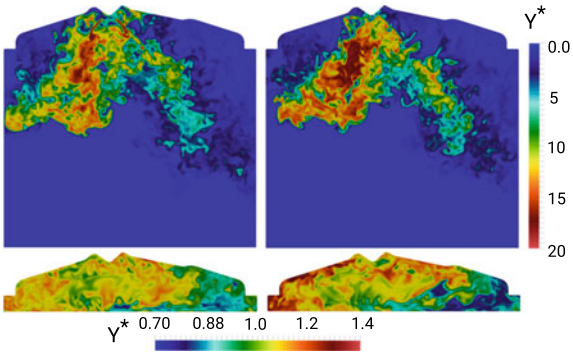


Fig. 9 Evaporated Ethanol (left) and 2-Butanone (right) fuel concentration in the tumble plane at 105° CA (top) and 340° CA (bottom). The fuel concentration is non-dimensionalized by the volume-averaged concentration at corresponding CA $Y^* = \frac{Y}{\widetilde{Y}}$

In Fig. 7 the parcels in the engine at two crank angles for both fuels are visualized. This comparison shows significant differences in spray propagation and number of parcels caused by the different fuel physical properties. A much larger impact on the spray development in the engine compared to the pressure chamber setup can be seen.

Due to the different evaporation rates (see Fig. 8), it takes about 1.6 times as long for Ethanol to evaporate half of the injected mass than for 2-Butanone. This results in a full evaporation at 116 CA for 2-Butanone and at 121 CA for Ethanol. For further illustration, the evolution of liquid and vapor mass fractions is displayed in Fig. 8.

As discussed in [4] and is also visible in Fig. 7 the liquid penetration on the intake port side is much lower than around the exhaust port side due to the interaction with the tumble vortex.

The difference in evaporation rate also affects the mixing of the gasouse fuels, which is qualitatively assessed for the instantaneous concentration in the tumble plane in Fig. 9. Due to the different fuel masses of Ethanol and 2-Butanone, the local concentration $Y(\mathbf{x}, t)$ has been non-dimensionalized by the volume mean concentration $\widetilde{Y(t)} = \oint_{V(t)} (Y(\mathbf{x}, t)/V(t))\, dV(t)$ to enable a better comparison.

Unlike the pressure chamber results (compare Fig. 4), a strong difference between the intake and exhaust side of the injector can be seen and the otherwise symmetrical cone shaped concentration deteriorates. The displacement effect of the open valves on the intake side leads to higher concentrations in that area. Additionally, the highly

three-dimensional flow field in the engine affects the concentration distribution. For all crank angles, 2-Butanone shows larger deviations from the mean concentration, i.e. richer regions with high local fuel concentrations.

Ethanol on the contrary has a lower evaporation rate, which leads to a higher penetration of the droplets in the cylinder volume. Thus, evaporation takes place in a larger engine volume leading to a more homogeneous mixing and less Ethanol droplets evaporate between the intake valves.

On the exhaust side, droplets are spread further apart by the predominant overflow of the intake jet which enhances mixture formation. At the end of the intake and during the compression stroke, mixing is primarily driven by the turbulence connected to the tumble vortex. Nevertheless, regions with larger variations of the local fuel concentration remain until the presumed start of ignition at 340° CA.

The evolution of the fuel distribution functions ($V(Y_i^*, t)$) during compression is determined by the computation of the volume ratio for cells with the concentration Y_i^*

$$V(Y_i^*, t) = \oint_{V(t)} \begin{cases} 100V(\mathbf{x}, t)/V(t) & \text{if} \\ Y_i^* - 0.005 < Y^*(\mathbf{x}, t) \leq Y_i^* + 0.005 & dV(t). \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

at 180° and 340° CA.

At bottom-dead center (Fig. 10 left), the Ethanol concentration distribution shows a more pronounced plateau around the average concentration ($Y_i^* = 1$) and smaller volumes with zero or rich concentrations. The extent of the plateau is a measure for a favorable homogeneous mixing. During the compression stroke a double peak evolves from the plateau with steeper shoulders for the Ethanol distribution compared to 2-Butanone. When analyzing the distribution functions at 340° CA (Fig. 10 right), the effect of different skewness of the two distribution functions can be seen. Skew-

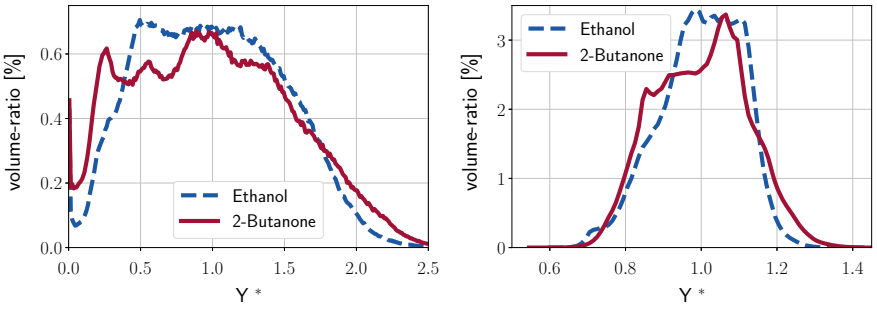


Fig. 10 Dimensionless fuel concentration distribution functions in the engine volume for 2-Butanone and Ethanol at 180° CA (left) and 340° CA (right). The volume ratios are computed based on a concentration step of 0.01 and are a percentage of the crank-angle dependent engine volume.

ness describes the asymmetry of a distribution function. Negative skewness denotes a left-tailed distribution function, in this application meaning that the fuel concentration has a longer tail in the lean fuel-air mixture and less regions with rich mixture. For positive skewness the tail direction is reversed. Whereas the 2-Butanone distribution is right-tailed during the entire compression stroke, Ethanol skewness changes sign at approximately 321° CA and becomes left-tailed. The left-tailed Ethanol distribution has far less rich outliers and a larger volume with concentrations slightly above the mean. The 2-Butanone distribution shows a larger volume for concentrations of 25–35% below the mean. For Ethanol, 62.1% of the engine volume has a difference of less than 10% from the target concentration, while this is the case for only 55.1% of the volume for 2-Butanone.

In conclusion, a significant effect of the duration of injection and the rate of evaporation on the fuel concentration distribution function can be seen. The unfavorable evaporation between the intake valves leads to a less homogeneous mixture and large variation from the mean concentration for 2-Butanone. The shape of the distribution function at 180° CA for Ethanol on the other hand leads to a slightly improved mixture formation at the start of ignition.

7 Conclusion

The mixture formation of Ethanol and 2-Butanone is predicted in a direct injection gasoline engine by performing an LES of the turbulent flow field coupled with validated spray models. The simulation methods can predict the difference in the fuel-air mixing formation caused by the varying fuel properties. While pressure chamber setups show no significant distinctions in the spray behavior, the mixing in the IC engine setup reveals larger differences. In agreement with previous experimental studies [4, 8], a higher evaporation rate of 2-Butanone compared to Ethanol caused by the higher fuel-air ratio and a lower specific heat of vaporization for 2-Butanone are observed. The fuel distribution at ignition crank angle are more favorable for a clean combustion for Ethanol compared to 2-Butanone, when using the same engine parameters. The Ethanol concentration distribution function shows less rich fuel regions compared to 2-Butanone. This differences is caused by a larger axial penetration of Ethanol due to its slower evaporation rate. Consequently, a better distribution of the fuel in the in-cylinder volume prevails during the intake and compression stroke. In the next step, a cost function based on the concentration distribution function will be defined, which will enable quantitative comparison of the mixing quality for subsequent optimization studies, i.e., with modified injection angles, positions and timings.

Acknowledgements Excerpts of this contribution have previously been published in [3]. The authors gratefully acknowledge the funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy- Cluster of Excellence 2186 "The Fuel Science Center" -ID: 390919832. The authors gratefully acknowledge the Gauss Centre for

Super-computing e.V. (www.gauss-centre.eu) for funding this project by providing computing time on the GCS Supercomputer HazelHen at HLRS Stuttgart.

References

1. J. Bellan, K. Harstad, The details of the convective evaporation of dense and dilute clusters of drops. Int. J. Heat Mass Transf. **30**(6), 1083–1093 (1987)
2. M.J. Berger, M. Aftosmis, S. Allmaras, Progress towards a Cartesian cut-cell method for viscous compressible flow, in *AIAA Paper* (2012), pp 2012–1301
3. S. Berger, T. Wegmann, M. Meinke, W. Schröder, Large-Eddy Simulation Study of Biofuel Injection in an Optical Direct Injection Engine. SAE Technical Paper 2020-01-2121 (2020) Injection in an Optical Direct Injection Engine. SAE Technical Paper 2020-01-2121 (2020)
4. M. Braun, J. Palmer, T.V. Overbrueggen, M. Klaas, R. Kneer, W. Schröder, Influence of in-cylinder air flow on spray propagation. SAE Int. J. Engines **10**, 1398–1410 (2017)
5. M. Dahmen, M. Hechinger, J.V. Villeda, W. Marquardt, Towards model-based identification of biofuels for compression ignition engines. SAE Int. J. Fuels Lubr. **5**(3), 990–1003 (2012)
6. C. Günther, M. Meinke, W. Schröder, A flexible level-set approach for tracking multiple interacting interfaces in embedded boundary methods. Comput. Fluids **102**, 182–202 (2014)
7. C. Günther, M. Meinke, W. Schröder, A flexible multiple level-set method for capturing and tracking moving embedded interfaces. Comp. Fluids **102**, 182–202 (2014)
8. F. Hoppe, U. Burke, M. Thewes, A. Heufer, F. Kremer, S. Pischinger, Tailor-made fuels from biomass: potentials of 2-Butanone and 2-Methylfuran in direct injection spark ignition engines. Fuel **167**, 106–117 (2016)
9. A. Lintermann, M. Meinke, W. Schröder, Zonal flow solver (ZFS): a highly efficient multi-physics simulation framework. Int. J. Comp. Fluid D. **34**, 458–485 (2020)
10. F. Mathieu, M. Reddemann, D. Martin, R. Kneer, Experimental investigation of fuel influence on atomization and spray propagation using an outwardly opening GDI-injector, in *SAE Technical Paper* (SAE International, 2010)
11. M. Meinke, L. Schneiders, C. Günther, W. Schröder, A cut-cell method for sharp moving boundaries in Cartesian grids. Comp. Fluids **85**, 134–142 (2013)
12. R.S. Miller, J. Bellan, Direct numerical simulation of a confined three-dimensional gas mixing layer with one evaporating hydrocarbon-droplet-laden stream. J. Fluid Mech. **384**, 293–338 (1999)
13. A. Niemöller, M. Schlottke-Lakemper, M. Meinke, W. Schröder, Dynamic load balancing for direct-coupled multiphysics simulations. Comput. Fluids **199**, 104437 (2020)
14. A. Pogorelov, L. Schneiders, M. Meinke, W. Schröder, An adaptive Cartesian mesh based method to simulate turbulent flows of multiple rotating surfaces. Flow Turbul. Combust. **100**(1), 19–38 (2018)
15. A. Putnam, Integratable form of droplet drag coefficient (1961)
16. R.D. Reitz, J. Beale, Modeling spray atomization with the Kelvin-Helmholtz Rayleigh-Taylor hybrid model. At. Spray **9**(6), 623–650 (1999)
17. L. Schneiders, C. Günther, M. Meinke, W. Schröder, An efficient conservative cut-cell method for rigid bodies interacting with viscous compressible flows. J. Comput. Phys. **311**, 62–86 (2016)
18. L. Schneiders, D. Hartmann, M. Meinke, W. Schröder, An accurate moving boundary formulation in cut-cell methods. J. Comput. Phys. **235**, 786–809 (2013)
19. J. Sim, J. Badra, A. Elwardany, H. Im, Spray modeling for outwardly-opening hollow-cone injector, in *SAE 2016 World Congress and Exhibition* (SAE International, 2016)
20. M. Thewes et al., Analysis of the impact of 2-Methylfuran on mixture formation and combustion in a direct-injection spark-ignition engine. Energy Fuels **25**(12), 5549–5561 (2011)
21. T. Van Overbrueggen, M. Braun, M. Klaas, W. Schröder, Experimental analysis of the impact of injected biofuels on in-cylinder flow structures. SAE Int. J. Engines **9**(2), 1320–1348 (2016)

Progress in IAG's Rotorcraft Simulation Framework

Manuel Keßler, Felix Frey, Johannes Letzgus, Constantin Öhrle, Jakob Thiemeier, and Ewald Krämer

Abstract The purpose of the project *HELISIM*, now under the auspices of Gauss Centre for Supercomputing (GCS), is the extension of high-fidelity aerodynamic, aeroelastic and aeroacoustic simulations of rotors and helicopters, both in model and full-scale configurations. Since the flow solver and the overall tool chain have reached a high level of maturity, algorithmic developments are no more the key focus of our projects, but occur as project and application requirements emerge. The helicopter & aeroacoustics group at IAG likewise concentrates on gaining physical insight, reducing risks in unconventional flight states and optimizing the aircraft itself in form of high-fidelity CFD simulations of most complex configurations. Simultaneously, accuracy, robustness and performance enhancements of the flow solver and the tool chain are always part of the work. All in all, the endeavor is to push the physical comprehension and the design and development of rotorcraft further.

1 Toolchain

All simulations described in this report utilized IAG's proven rotorcraft simulation framework, centering around the FLOWer fluid flow solver, originally developed by DLR [1] and significantly extended by IAG for aerodynamic Helicopter simulations [2, 3] as well as aeroacoustic analyses [4]. Starting as a standard multi-block structured finite volume solver for the unsteady Reynolds averaged Navier-Stokes equations (URANS), it evolved into a highly sophisticated tool with all capabilities required for advanced rotorcraft analysis, including Chimera technology of overlapping grids, mesh motion and deformation, and high order discretisations up to 6th order WENO [5]. Over the last two decades the framework has been supplemented

M. Keßler (✉) · F. Frey · J. Letzgus · C. Öhrle · J. Thiemeier · E. Krämer
Institute of Aerodynamics and Gas Dynamics (IAG), University of Stuttgart, Pfaffenwaldring 21, 70569 Stuttgart, Germany
e-mail: kessler@iag.uni-stuttgart.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_21

by the comprehensive codes HOST [6], developed by Airbus Helicopters (formerly Eurocopter), and CAMRAD II [7] from Johnson Aeronautics, used for blade structure dynamics, and flight mechanics to reach a trimmed state in free flight, adjusting helicopter attitude and control angles. Over this time, many tools supporting the framework have been implemented according to actual needs, for local and global loads evaluation, sophisticated background grid generation, blade meshing, and structured post-processing. Recent additions include automatic grid refinement [5] and a library to handle tight coupling for structure deformations at the airframe [8].

Those supporting tools are clearly inevitable for productive research within the framework, although nearly 100% of the computing resources are consumed by the flow solver itself. Typical setup sizes range between about 10 million cells for component analysis, as a single propeller, and more than a billion cells for highly resolved wake vortex investigations. Typical full configurations of entire rotorcraft including all components appropriate for common research activities comprise around 200 million cells in 100 grid structures. FLOWer is fully parallelized and shows excellent weak scaling demonstrated up to 100000 cores and strong scaling below 10000 cells per core, as much as permitted by the grid splitting [9]. In the latter case, even for ideal blocking the number of communicated ghost cells is of the same size as the number of computed physical cells, marking more or less the theoretical limit of strong scaling.

Functional development of the framework is driven by current projects' requirements. For example, several shielding methods for Detached Eddy Simulation (DES) were investigated and implemented [10, 11]. The rotating geometry of helicopters with their extremely complex flow field with separation zones and backflow poses significant challenges on the generality of such methods to separate URANS regions from LES parts reliably. Several innovative ideas were devised, implemented, tested and assessed. Another path of development added trim capabilities for additional degrees of freedom, like flaps. Although not strictly necessary to reach a stationary flight state, they open up room for improved operations, like reduced power, vibrations, or noise. Finally, initial maneuver flight capabilities were added, in order to conquer the current barrier of stationary flight cases and proceed into the realm of non-inertial flight conditions, as are very common for rotorcraft with their unique maneuver aptitude.

Performance improvements, on the other hand, are taken on mostly on a sideline if and only if a specific case shows deficiencies in one phase or another. For example, at complex configurations the Chimera evaluation may slow down the computation, and then some effort is spent to speed up this process. In another setup, load evaluation proved suboptimal, so the respective module was analyzed specifically and improved. By this demand-driven approach, optimizing work is always directed to parts where it pays off most. However, due to the current transition from the former Hazel Hen hardware to the new Hawk architecture, some more advanced tuning is expected and already happening in order to utilize the machine's hardware potential at its best. Initial experiments (see Fig. 1) on Hawk seem to indicate that memory bandwidth poses a bottleneck for the much larger number of cores in a node, which may require substantial reorganization of data handling.

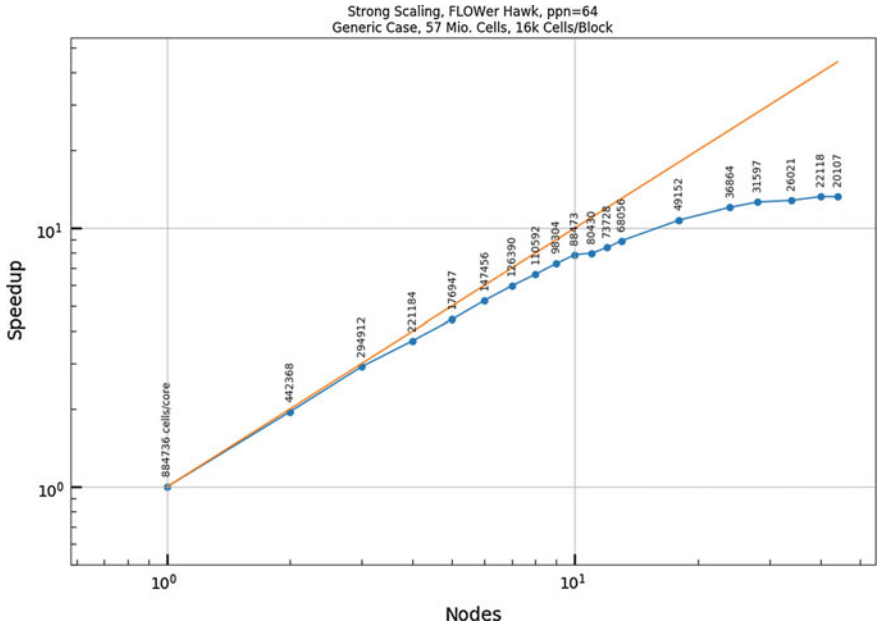


Fig. 1 Strong scaling of smaller FLOWer case on the new Hawk system

2 Dynamic Stall on a Full-Scale Helicopter in Turn Flight

Dynamic stall is one of the most complex flow phenomena in rotorcraft aerodynamics and it limits the flight envelope of a helicopter. Thus, a highly loaded, high-speed turn flight—which was classified as dynamic-stall relevant—of Airbus Helicopters' *Bluecopter* demonstrator helicopter was numerically investigated with a loose coupling of the CFD code FLOWer and the rotorcraft comprehensive code CAMRAD II, performing a three-degree-of-freedom trim of the isolated rotor. On the CFD side, a high-fidelity DDES was carried out and differences between the SA-DDES and SST-DDES turbulence model as well as the influence of elastic twist (elastic versus torsionally rigid rotor), of additionally modeled helicopter components (hub, fuselage and tail boom) and of a variation of the target rotor thrust (low, mid and high) were investigated. Computed rotor control angles and pitch-link loads were then compared to flight-test measurements.

The FLOWer computations with domain sizes of 158 to 165 million grid cells (Fig. 2) were carried out on the Hazel Hen cluster of HLRS, where the grids, split into 7700 blocks, were computed on 7000 cores. For one trim step a wall time of roughly 6.5 h was required.

The rotor flow field in the present flight condition is highly unsteady and complex, as the flow phenomena of trailing-edge separation, blade-vortex interaction (BVI), several different dynamic-stall events and shock-induced separation occur throughout

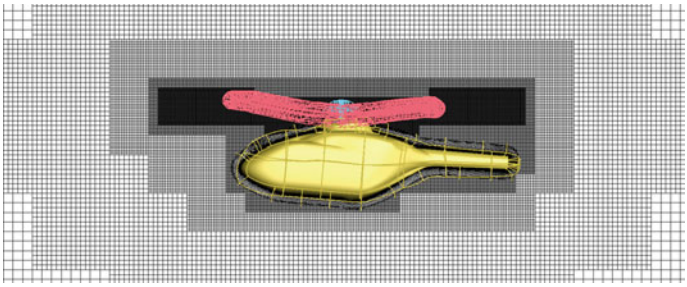


Fig. 2 Section of the Cartesian off-body grid with levels of refinement adapted to the body grids.

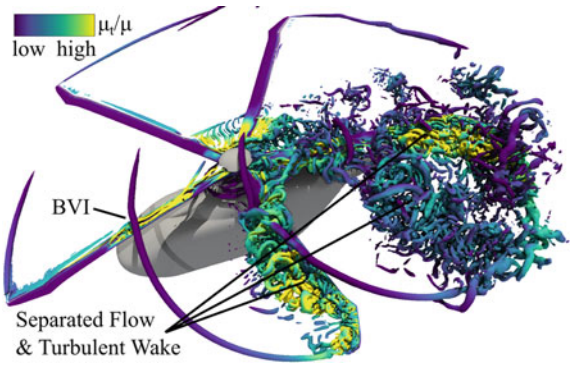


Fig. 3 Visualization of the instantaneous flow field by means of isosurfaces of the λ_2 criterion colored by the viscosity ratio μ_t/μ.

a large portion of the azimuth, see Fig. 3. The first dynamic-stall event happens in the third quadrant of the rotor disk and might be triggered by blade-vortex interaction. The dynamic-stall vortex quickly spreads in- and outboards, inducing overshoots of sectional thrusts and pitching moments. The kink in the planform of the rotor blade seems to delay the outboard spreading of dynamic stall and is the origin of a tip-vortex-like apex vortex. Several other typical dynamic-stall events and streamwise vortices appear in the outboard region of the blade in the fourth quadrant, which sustain overshoots of sectional thrust and pitching moments.

With the SA-DDES turbulence model, the flow separation in the fourth quadrant of the rotor disk is less pronounced than with SST-DDES, leading to an overall worse agreement with the flight test. The additional consideration of the hub cap, fuselage with mast fairing and the tail boom in CFD alters the rotor inflow, which improves the agreement with both flight-test control angles and pitch-link loads, see Fig. 4. The control angles of all cases converge reasonably within at least six loosely coupled CFD/CSD iterations. While the lower target thrust cases underpredict and the higher target thrust case overpredicts the flight-test control angles, the control angles of the mid target thrust agree very well with the measurements. All computed CFD/CSD cases underpredict the amplitudes of the flight-test pitch-link loads and phase shifts are apparent. However, overall trends agree reasonably and it becomes clear that CFD/CSD coupling and trimming yields better results than a standalone CAMRAD II

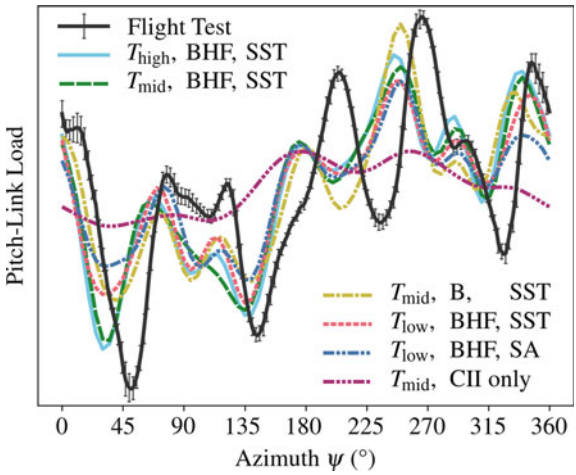


Fig. 4 Comparison of measured and computed pitch-link loads in the time domain.

analysis. Most likely causes for the remaining deviations are deficiencies in meeting the free-flight trim conditions or in the structural-dynamic modeling of the main rotor; therefore, as a next step, the elasticity of the rotor controls is to be considered.

3 Aeromechanics of Compound Helicopters

3.1 Parameter Variation of Redundant Helicopter Controls

Within the CA^{3}TCH project the utilization of redundant helicopter controls is investigated. The compound helicopter RACER features overall 13° of freedom (see Fig. 5), six more than a conventional helicopter. However, there are only six constraints in a steady flight: $\sum F = 0$ and $\sum M = 0$. Hence, the trim problem is overdetermined by seven degrees of freedom for this rotorcraft. The remaining degrees of freedom can redundantly be used to satisfy additional trim constraints, e.g. a specific lift share between wings and main rotor or a specific propulsion share between propellers and main rotor. This allows a variation of a given trim for one single flight condition.

In a broad parameter study 13 different flight physical quantities like lift or propulsion share have been identified and varied. This was done by means of the established coupling infrastructure at IAG consisting of the comprehensive analysis tool HOST and the CFD solver FLOWer as main components. The selected flight condition is a horizontal cruise flight at 175 knots which is at the upper advance ratio range of conventional helicopters. This flight state features typical phenomena at the main rotor under high advance ratio but leaves sufficient opportunities for trim variations without exceeding extreme trim conditions. The objective of this study is a deeper

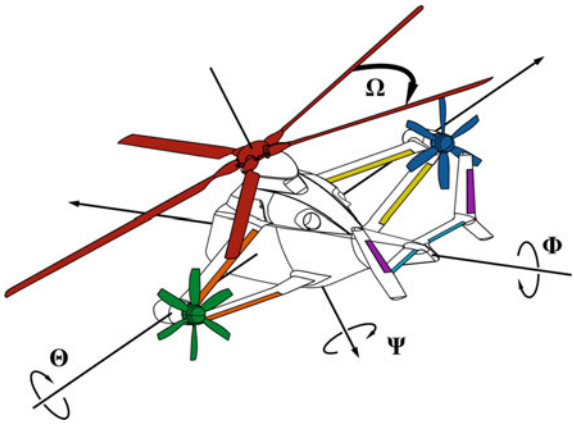


Fig. 5 Visualization of the available degrees of freedom of RACER. Coloured parts represent different controls. Additional degrees of freedom, e.g. attitude angles and main rotor speed, are shown.

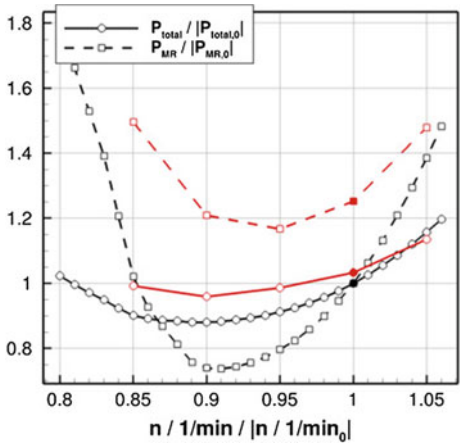


Fig. 6 Power consumption over the variation of main rotor speed. Main rotor power and total system power shown.

understanding of the interaction between redundant flight controls in order to derive findings concerning for example an increase in efficiency.

HOST provides flight mechanic estimations on a comprehensive but low fidelity basis. These comprehensive results can be assessed by the comparison with high fidelity results provided by FLOWer. Figure 6 shows exemplary results of the variation of main rotor speed. FLOWer results are represented by black colour, whereas HOST results are printed in red. All shown data points are normalized by the respective quantities of the baseline case computed by HOST and marked by the black filled symbol. HOST clearly underestimates both the main rotor and total system power. However, the agreement for total power is better than for main rotor power. Also, the minimum of the total power at $\frac{n}{n_0} \approx 0.9$ matches for both methods. However, this is not the case for the main rotor power which shows a shift of $\Delta\frac{n}{n_0} \approx 0.05$.

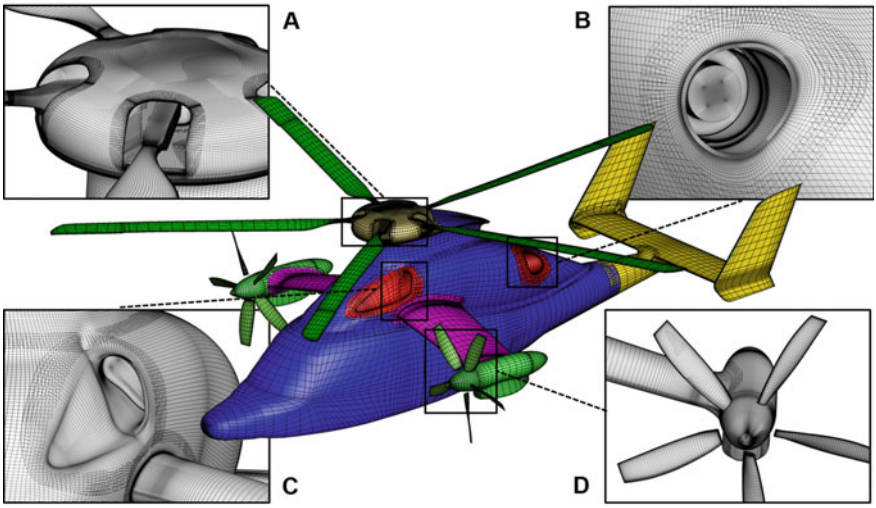


Fig. 7 CFD simulation setup for the X^3 compound helicopter.

3.2 Validation of the Aeromechanic Simulations

A necessary and also very important step for the simulation of new rotorcraft concepts is the validation and the estimation of error margins of the simulation methods applied. For complete helicopter simulations in a free-flight condition, a possibility to validate the methods is to correlate the predictions with flight test data. Therefore, an extensive validation study for the compound helicopter X^3 has been performed. In this context, overall three different flight states have been simulated and compared with the flight test data. The flight conditions range from hover over cruise to high-speed dive.

Figure 7 shows the CFD simulation setup of the X^3 for the flow solver FLOWer that consists of around $70 - 150$ million grid cells, depending on the flight state. For all simulation cases, an iterative loose coupling between FLOwer and HOST is applied. The comparison of the simulation results with the flight test consists of performance parameters, flight mechanics parameters such as trim controls, and rotor dynamic data. Overall, the applied simulation methods are validated for a compound helicopter configuration in different flight conditions. An exemplary comparison is shown in Figs. 8 and 9, comparing the flap bending moments of the rotor blade between simulation and flight test.

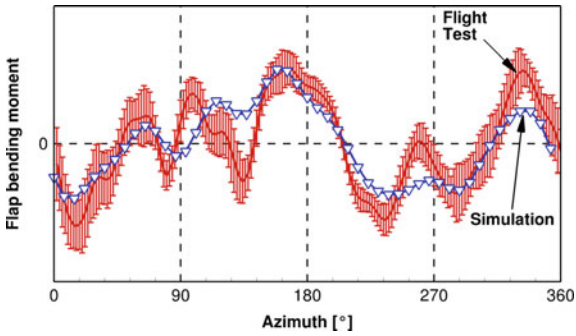


Fig. 8 Comparison of flap bending moment at 14% span.

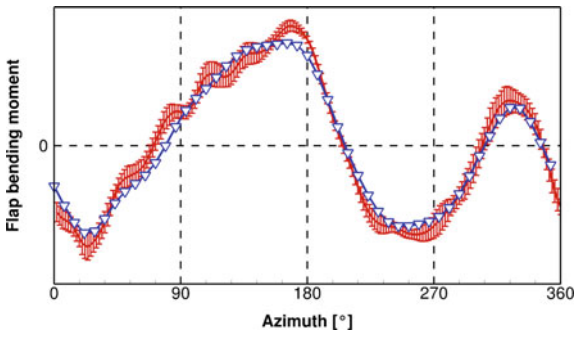


Fig. 9 Comparison of flap bending moment at 38% span.

3.3 Aeroelastic Simulation of a Compound Helicopter

Depending on the flight condition of a rotorcraft, significant aerodynamic interactions between the unsteady flow field and the airframe might occur. If the excitation frequency of the flow occurs near an eigenfrequency of the airframe, resonance is possible. This phenomenon is also called tail shake. As the compound helicopter Racer uses an H-tail configuration, an aeroelastic simulation offers significant insight into the dynamic excitation of the tail. In order to investigate this, a coupled CFD-FEM method is applied (see Fig. 10). Within this method, the CFD solver FLOWer computes the airloads which are transferred to a structural dynamics solver in each physical time step. The structural dynamics solver computes the resulting deformations that are subsequently transferred back to the CFD solver.

The capability was implemented for a conventional helicopter configuration and successfully applied to a (mild) tail shake flight condition [8]. Although the computational effort is huge due to the necessity of simulating tens of revolutions to obtain meaningful statistics for low frequency modes, the ability to predict this phenomenon at all before flight tests is an excellent improvement over trial and error.

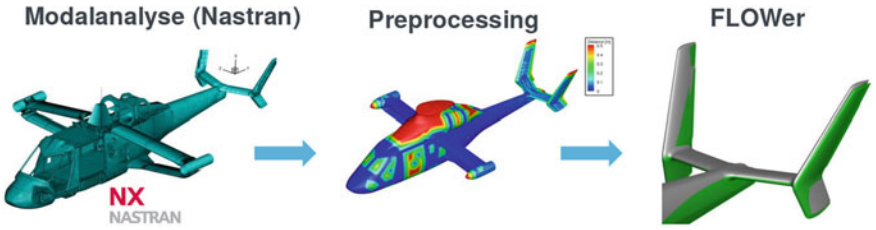


Fig. 10 Coupled FEM-CFD simulation method of the compound helicopter RACER.

Overall three different flight conditions have been investigated for the compound configuration. The simulation setup size is 200 million cells. Again, due to the fact that the relevant airframe eigenfrequencies have a relatively low frequency, a very long simulation time is required in order to resolve the aeroelastic behaviour. Therefore, approximately 100 revolutions were computed for each case.

3.4 Maneuvering Helicopter

Maneuver simulation capability was added to the framework with the intent of application to a model helicopter, for which flight testing was planned as an initial means of validation. Consequently, motion specification and structure modeling was added, and the configuration tested. Due to circumstances, unfortunately this flight testing had to be postponed, but in order to utilize the functionality implemented in the solver and to further the remaining framework we switched to a different setup, comprising a full-scale Robinson R44 type, created recently within two student's theses. Fortunately, the manufacturer of this helicopter, Robinson Inc. was very supportive delivering necessary data not easily measureable like blade stiffness, rotational inertia and even prepared CAD geometry for reference. In consequence, the setup was up and running at the time the model helicopter simulations were put back, so we took the opportunity to start applying the developing solver features to this different model.

For a start, transition from hover to forward climb flight was prescribed at a fixed trajectory, according to a generic flight pattern. The correct definition of the virtual flight path proved somewhat challenging and required a couple of extensions, but in the end the maneuver was represented fairly well. Figure 11 shows some snapshots of the flow field during this transition.

Another maneuver more critical from a flight dynamics point of view was the simulation of a quick stop with rapid deceleration from full forward flight to hover by strong pitch up. As no flight path data was available for either case, a helicopter simulator was utilized to create a realistic position, velocity and attitude schedule for this maneuver. Again, the simulation framework was extended to finally model this task as well, as seen in the snapshots of Fig. 12.

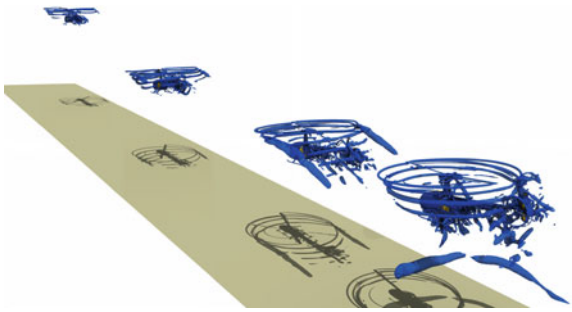


Fig. 11 Transition from hover to climb

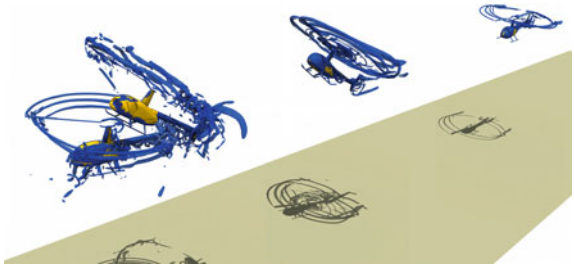


Fig. 12 Snapshots of quick stop maneuver

Every single run is extremely challenging due to the large number of revolutions necessary (about 100) to match real times of some 10 s at typical revolution speeds of 400 RPM, necessary to represent a realistic maneuver. Even with a somewhat simplified setup of 20 million cells only, parallelization is limited to about 15000 cells per core to still be efficient. This strongly urges to improve strong scaling properties on the new Hawk hardware with more cores per node and a (relatively) weaker interconnect, to be taken on in the upcoming reporting period.

4 Conclusions

Despite the extremely challenging geometry, mechanics, and flow physics, rotorcraft simulation has reached a level of confidence and accuracy where the focus slowly shifts from implementation to application. Specific investigations on advanced flow physics, complex fluid-structure coupling, or sophisticated flight mechanics allow a better understanding of fundamental phenomena, at the complexity level of full configurations. The detail provided by temporal flow field data allows the deep analysis of interaction phenomena characteristic for rotorcraft. This complements flight testing at eye level, but without the inevitable impediments of instrumentation and reproducibility. IAG's rotorcraft simulation framework is here at the international forefront of applied aeromechanic research for an extremely challenging engineering application.

Acknowledgements The provided supercomputing time and technical support of the High Performance Computing Center Stuttgart (HLRS) of the University of Stuttgart within the *HELISIM* project is gratefully acknowledged. Parts of the research presented in this study were supported by Deutsche Forschungsgemeinschaft (DFG) within the project *Untersuchung der dreidimensionalen dynamischen Strömungsablösung an Rotorblättern.* Another part was performed within the European research framework Clean Sky 2 under Grant 686530: *Coupled Aerodynamic Aeroacoustic Analysis of a Trimmed Compound Helicopter (CA^3TCH).*

References

1. J. Raddatz, J.K. Fassbender, Block structured Navier-Stokes solver FLOWer, in ed. by N. Kroll, J.K. Fassbender. *MEGAFLOW - Numerical Flow Simulation for Aircraft Design* (Springer, Heidelberg, Germany, 2005), pp. 27–44
2. U. Kowarsch, C. Öhrle, M. Hollands, M. Keßler, E. Krämer, Computation of helicopter phenomena using a higher order method (Springer, Cham, 2014), pp. 423–438
3. P. Kranzinger, U. Kowarsch, M. Schuff, M. Keßler, E. Krämer, Advances in parallelization and high-fidelity simulation of helicopter phenomena (Springer, Cham, 2016), pp. 479–494
4. E. Rebecca Busch, M.S. Wurst, M. Keßler, E. Krämer, Computational aeroacoustics with higher order methods, in ed. by W.E. Nagel, D.H. Kröner, M.M. Resch. *High Performance Computing in Science and Engineering 2012* (Springer Verlag, 2013)
5. C. Öhrle, U. Schäferlein, M. Keßler, E. Krämer, Higher-order simulations of a compound helicopter using adaptive mesh refinement, in *American Helicopter Society 74th Annual Forum* (2018)
6. B. Benoit, A.-M. Dequin, K. Kampa, W. von Grünhagen, P.-M. Basset, B. Gimonet, HOST, a general helicopter simulation tool for Germany and France, in *American Helicopter Society, 56th Annual Forum, Virginia Beach, Virginia* (2000)
7. W. Johnson, CAMRAD II comprehensive analytical model of rotorcraft aerodynamics and dynamics, 4.8 edn (2009)
8. U. Schäferlein (né Kowarsch), M. Keßler, E. Krämer, Aeroelastic simulation of the tail shake phenomenon. J. Am. Helic. Soc. **63**(3), 1–17 (2018)
9. J. Letzgus, L. Dürrwächter, U. Schäferlein, M. Keßler, E. Krämer, Optimization and HPC-applications of the flow solver FLOWer (Springer, Cham, 2018), pp. 305–322
10. P. Weihing, J. Letzgus, T. Lutz, E. Krämer, Development of alternative shielding functions for detached-eddy simulations, in *Proceedings of the 7th Symposium on Hybrid RANS-LES Methods* (2018)
11. J. Letzgus, A.D. Gardner, T. Schwermer, M. Keßler, E. Krämer, Numerical investigations of dynamic stall on a rotor with cyclic pitch control. J. Am. Helic. Soc. **64**(1), 1–14 (2019)

Increasing the Flexibility of the High Order Discontinuous Galerkin Framework FLEXI Towards Large Scale Industrial Applications

Andrea Beck, Min Gao, Daniel Kempf, Patrick Kopper, Nico Krais, Marius Kurz, Jonas Zeifang, and Claus-Dieter Munz

Abstract This paper summarizes our progress in the application and improvement of a high order discontinuous Galerkin (DG) method for scale resolving fluid dynamics simulations towards robust and flexible industrial applications. We report the results obtained on the Cray XC40 Hazel Hen cluster at HLRS and show code performance. We present three application cases and developments: An implicit time integration scheme for split-form DG schemes allows us to solve stiff problems with increased efficiency, which will open up new classes of problems for simulations with FLEXI. We follow this by discussing a Large Eddy Simulation (LES) of a compressible turbulent boundary layer and provide comparison to DNS data. Lastly, we demonstrate how to extend the high order scheme with a consistent and conservative sliding mesh interface, and present results of a 1.5 stage turbine simulation with wall-resolved LES.

M. Gao · D. Kempf · N. Krais · M. Kurz · J. Zeifang · C.-D. Munz
Institute of Aerodynamics and Gas Dynamics, University of Stuttgart, Stuttgart, Germany
e-mail: mg@iag.uni-stuttgart.de

D. Kempf
e-mail: kempf@iag.uni-stuttgart.de

N. Krais
e-mail: krais@iag.uni-stuttgart.de

M. Kurz
e-mail: m.kurz@iag.uni-stuttgart.de

J. Zeifang
e-mail: zeifang@iag.uni-stuttgart.de

C.-D. Munz
e-mail: munz@iag.uni-stuttgart.de

P. Kopper
Institute of Aircraft Propulsion Systems, University of Stuttgart, Stuttgart, Germany
e-mail: kopper@ila.uni-stuttgart.de

A. Beck (✉)
Laboratory of Fluid Dynamics and Technical Flows, Otto-von-Guericke University Magdeburg, Magdeburg, Germany
e-mail: andrea.beck@ovgu.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_22

1 Introduction

The simulation framework FLEXI has been developed during the last years at the working group of Prof. Munz. The spatial discretization is based on an arbitrary high order discontinuous Galerkin spectral element method (DGSEM); different schemes are available for temporal integration. FLEXI is focused on scale resolving simulations of compressible flow and associated problems like aeroacoustics and particle-flow-interactions. A very recent, comprehensive overview of FLEXI, its features, structure and supporting environment can be found in [17]; a number of successful large scale simulations with FLEXI on systems at HLRS can be found e.g. in [1–6, 9]. In this report, we present some novel developments, examples and extensions of FLEXI towards further realistic applications of multiphysics problems. We first start in Sect. 2 by discussing the extension of the current time integration options towards implicit methods. In implicit time integration schemes, the time step restriction that often plagues explicit schemes in regions of stiffness is lifted, leading to time steps that are merely limited by physical considerations and not numerical stability requirements. This advantage comes at the price of having to solve large, typically non-linear systems of equations for each time step. The solution procedure for these systems often includes a preconditioner, which naturally has to be adapted to the system matrix. In Sect. 2, we show our results in the derivation of such a preconditioner for split form DGSEM, as well as a first application of the resulting implicit scheme to an aeroacoustic problem. We discuss the performance in contrast to an explicit scheme and highlight the influence of the time step on the solution accuracy. In Sect. 3, we investigate the LES of a turbulent, compressible boundary layer, which will be used in the future to assess fluid structure interaction of shocks and compliant walls. In the last part, we will discuss the feature extension of FLEXI towards turbomachinery applications and therefor present a conservative high order sliding mesh method (Sect. 4). While the consistent inclusion of new features into a high order scheme is a challenge in itself, doing that while keeping the parallel performance and scalability stable is a non-trivial task. In particular, new features that introduce an inherent load imbalance need to be investigated carefully. We will thus also present scaling and parallel performance data for the sliding mesh approach, and conclude the section by showing large scale simulation results obtained on Hazel Hen. We end with a brief summary and some preliminary glimpse at the porting of FLEXI to HAWK.

2 Implicit Time Discretization for the Split DG Formulation

In this section, we briefly summarize the necessary building blocks for implicit time integration schemes for FLEXI. The split DG implementation is based on the strong form DGSEM on Gauss-Lobatto nodes. Following the notation from [13], we first

introduce the spatial semi-discrete formulation of the DGSEM. Then, the temporal discretization with implicit Runge-Kutta methods is presented.

2.1 Split Form DGSEM

In our simulations we consider the compressible Navier-Stokes equations with an ideal gas equation of state in the usual notation, where $\mathbf{U}$ denotes the vector of conserved quantities $\mathbf{U} = (\rho, \rho\mathbf{u}, \rho e)^T$ and F^{hyp} and F^{visc} the associated convective and viscous flux vectors. In the interest of space, we present the method in 2D. The split formulation of DGSEM is used to control the aliasing errors, introduced by the nonlinearity of the advection terms. Split forms can be seen as a correction for the product rule on the discrete level [11]. They use a substitution of the volume integral with a special version using two-point volume flux functions $\boldsymbol{\mathcal{F}}^{\#}$. The DGSEM uses nodal Lagrange interpolation polynomials ℓ_i with degree N to approximate the solution and the fluxes. Integrals are approximated by using the Gauss-Lobatto quadrature rule. A collocation of the interpolation and quadrature nodes leads to the DGSEM formulation. With this, the semi-discrete version of the split form DGSEM according to [11] in two dimensions reads as

$$\begin{aligned}\frac{\partial \mathbf{U}_{ij}}{\partial t} = \\ -\frac{1}{\mathscr{J}_{ij}}\Bigg[\sum_{\alpha=0}^{N} \boldsymbol{\mathcal{F}}^{\#,1}(\mathbf{U}_{ij}, \mathbf{U}_{\alpha j})\tilde{D}_{i\alpha} + \left(\left[f^*\hat{s}\right]_j^{+\xi^1}\hat{\ell}_i^+ + \left[f^*\hat{s}\right]_j^{-\xi^1}\hat{\ell}_i^-\right) \\ + \sum_{\beta=0}^{N} \boldsymbol{\mathcal{F}}^{\#,2}(\mathbf{U}_{ij}, \mathbf{U}_{i\beta})\tilde{D}_{j\beta} + \left(\left[f^*\hat{s}\right]_i^{+\xi^2}\hat{\ell}_j^+ + \left[f^*\hat{s}\right]_i^{-\xi^2}\hat{\ell}_j^-\right)\Bigg] \forall\, i, j,\end{aligned}$$

where $\boldsymbol{\mathcal{F}}^{\#,\cdot}$ denotes the two-point flux in reference coordinates, $f^*\hat{s}$ the numerical flux with its corresponding surface element, $\mathscr{J}$ the Jacobian of the transformation from physical to reference space, D the differentiation matrix $D_{ij} := \partial\ell_j(\xi)/\partial\xi|_{\xi=\xi_i}$ and $\hat{\ell}_i^{\pm}$ approximates $\hat{\ell}_i^{\pm} = \ell_i(\pm 1)/\omega_i$, with ω_i denoting the i-th weight of the quadrature rule. The modified differentiation matrix $\tilde{D}$ is given by

$$\tilde{D}_{00} = D_{00} + \frac{1}{2\cdot\omega_0}, \quad \tilde{D}_{NN} = D_{NN} - \frac{1}{2\cdot\omega_N}.$$

This modified D matrix automatically takes the surface integral over the inner flux into account. The $1/2$ comes from the fact that we incorporated the factor of 2 in the split fluxes, but need to recover the Euler fluxes here. $1/\omega_{0/N}$ then simply is $\hat{\ell}(\mp 1)$. This means that the surface integral is the same as it is used for the weak form of the DGSEM, see e.g. [13], if Gauss-Lobatto nodes are used. With this, for the modified differentiation matrix $\tilde{D}_{ii} = 0\ \forall i$ holds.

2.2 Implicit Time Discretization for DGSEM

For the temporal discretization, implicit Runge-Kutta methods are used. We are following the presentation in [25], where explicit first stage singly diagonal implicit Runge-Kutta (ESDIRK) schemes have been applied to the DGSEM.

The temporal semi-discrete formulation for globally stiffly accurate ESDIRK schemes, such as introduced in [15], with s stages is given by

1. For $i = 1, \ldots, s$ solve for $\mathbf{U}^{n,i}$

$$\mathbf{U}^{n,i} - \mathbf{U}^n + \Delta t \sum_{j=1}^{i} a_{ij} \mathbf{R}(\mathbf{U}^{n,j}, \nabla_x \mathbf{U}^{n,j}, t^n + c_j \Delta t) = 0. \tag{1}$$

2. Set $\mathbf{U}^{n+1} := \mathbf{U}^{n,s}$,

with the specific scheme dependent coefficients a_{ij} and c_j and the spatial operator $\mathbf{R}$. To solve the non-linear equations given in Eq. (1), Newton's method is used. This leads to the necessity of solving a linear system in each Newton iteration. For that purpose, the iterative GMRES method [23] is applied in each Newton iteration k:

$$\frac{d\mathbf{f}(\mathbf{U}^k)}{d\mathbf{U}^k} \Delta \mathbf{U}^k = -\mathbf{f}(\mathbf{U}^k), \quad \text{with} \tag{2}$$

$$\mathbf{f}(\mathbf{U}^k) = (\mathbf{I} - \Delta t a_{i,i} \mathbf{R}) \mathbf{U}^k - \mathbf{U}^n + \Delta t \sum_{j=1}^{i-1} a_{ij} \mathbf{R}(\mathbf{U}^{n,j}, \nabla_x \mathbf{U}^{n,j}, t^n + c_j \Delta t). \tag{3}$$

A Jacobian-free approach, see e.g. [16], is chosen to evaluate the matrix-vector product in Eq. (2). To accelerate the convergence properties of the linear solver, a preconditioner can be applied. The purpose of the preconditioner is to approximate $d\mathbf{f}(\mathbf{U}^k)/d\mathbf{U}^k$ as accurately as possible, but at the same time being much easier to invert. In [25] an element-wise block-Jacobi preconditioner for the standard DGSEM has been presented. It considers the dependencies of degrees of freedom inside the same element and neglects inter-element dependencies. In the following section, the block-Jacobi preconditioner for the split form DGSEM is derived.

2.3 Block-Jacobi Preconditioner for the Split Form DGSEM

As the split formulation only changes the hyperbolic flux contribution, the reader is referred to [25] for the preconditioner of the viscous flux and only the derivatives of $\mathbf{R}$ with respect to $\mathbf{U}$ are shown here. First, the contribution of the surface integral is given by

$$-\mathscr{J}_{ij}\frac{d\mathbf{R}_{ij}^{\text{surface}}}{d\hat{\mathbf{U}}_{mn}} = \frac{\partial}{\partial\hat{\mathbf{w}}_{mn}}\Bigg[\left([\mathbf{f}^*\hat{s}]_j^{+\xi^1}\hat{\ell}_i^+ + [\mathbf{f}^*\hat{s}]_j^{-\xi^1}\hat{\ell}_i^-\right) + \left([\mathbf{f}^*\hat{s}]_i^{+\xi^2}\hat{\ell}_j^+ + [\mathbf{f}^*\hat{s}]_i^{-\xi^2}\hat{\ell}_j^-\right)\Bigg]. \tag{4}$$

For example, the derivative of the flux at the right surface $+\xi^1$ can be calculated as

$$\frac{\partial[\mathbf{f}^*\hat{s}]_j^{+\xi^1}}{\partial\hat{\mathbf{w}}_{mn}}\hat{\ell}_i^+ = \frac{\partial[\mathbf{f}^*\hat{s}]_j^{+\xi^1}}{\partial\hat{\mathbf{w}}_{Nj}}\hat{\ell}_i^+\delta_{im}\delta_{jn},$$

where the derivative of the numerical flux is calculated via a first order finite difference to ensure a broad applicability of different Riemann solvers.

For the remaining volume integral contribution of the derivative $d\mathbf{R}_{ij}^{\text{volume}}/d\mathbf{U}_{mn}$, the structure of the two-point fluxes has to be considered. In contrast to the standard DGSEM, the two-point fluxes introduce additional dependencies, such that more entries have a contribution. The derivation leads to

$$\begin{aligned}-\mathscr{J}_{ij}\frac{d\mathbf{R}_{ij}^{\text{volume}}}{d\mathbf{U}_{mn}} &= \\ \frac{\partial\boldsymbol{\mathcal{F}}^{\#,1}(\mathbf{U}_{mn},\mathbf{U}_{in})}{\partial\mathbf{U}_{mn}}\tilde{D}_{im}\delta_{jn} + \sum_{l=0}^{N}\frac{\partial\boldsymbol{\mathcal{F}}^{\#,1}(\mathbf{U}_{ln},\mathbf{U}_{mn})}{\partial\mathbf{U}_{mn}}\tilde{D}_{ml}\delta_{im}\delta_{jn} + & \\ \frac{\partial\boldsymbol{\mathcal{F}}^{\#,2}(\mathbf{U}_{mn},\mathbf{U}_{mj})}{\partial\mathbf{U}_{mn}}\tilde{D}_{jn}\delta_{im} + \sum_{l=0}^{N}\frac{\partial\boldsymbol{\mathcal{F}}^{\#,2}(\mathbf{U}_{ml},\mathbf{U}_{mn})}{\partial\mathbf{U}_{mn}}\tilde{D}_{nl}\delta_{im}\delta_{jn}, &\end{aligned} \tag{5}$$

where the sum over l was introduced by the two-point fluxes. The partial derivatives of the split fluxes themselves depend on the chosen split formulation. Note that while the main diagonal of $\tilde{D}$ is zero, some of the entries in the sum vanish. The block-Jacobi preconditioner can then be assembled by the sum of Eqs. (4) and (5). The inversion of the block-Jacobi preconditioner is done by LU-decomposition.

2.4 Application of the Implicit Split for DGSEM to Aeroacoustics

With the implicit-in-time split DGSEM formulation in place, applications to e.g. aeroacoustics are possible. To illustrate the capabilities and restrictions of the implicit time discretization, the flow over a cavity is considered.

2.4.1 Split Flux According to Pirozzoli

First, a suitable formulation of the two point fluxes has to be chosen. We choose the split flux formulation according to [20], which can be formulated as

$$\boldsymbol{\mathcal{F}}^{\#} = \begin{pmatrix} 2\{\rho\}\{\mathbf{u}\} \\ \boldsymbol{\mathcal{F}}_1^{\#}\{\mathbf{u}\} + 2\{p\}\mathbf{I} \\ \boldsymbol{\mathcal{F}}_1^{\#}\{H\} \end{pmatrix},$$

where $\{\cdot\}$ denotes the arithmetic mean and H being the total Enthalpy $H = \frac{\rho e + p}{\rho}$. The derivatives of the split flux formulation required for Eq. (5) in the x-direction can be derived as

$$\frac{\partial \boldsymbol{\mathcal{F}}^{\#,1}(\mathbf{U}, \mathbf{U}_{\text{ref}})}{\partial \mathbf{U}} = \begin{pmatrix} \frac{1}{2}(u_{\text{ref}} - \rho_{\text{ref}}\frac{u}{\rho}) & \frac{1}{2}(1 + \frac{\rho_{\text{ref}}}{\rho}) \\ \frac{\partial \mathscr{F}_1^{\#,1}}{\partial \rho}\{u\} - \mathscr{F}_1^{\#,1}\frac{u}{2\rho} + \frac{\partial p}{\partial \rho} & \frac{\partial \mathscr{F}_1^{\#,1}}{\partial \rho u}\{u\} + \frac{\mathscr{F}_1^{\#,1}}{2\rho} + \frac{\partial p}{\partial \rho u} \\ \frac{\partial \mathscr{F}_1^{\#,1}}{\partial \rho}\{v\} - \mathscr{F}_1^{\#,1}\frac{v}{2\rho} & \frac{\partial \mathscr{F}_1^{\#,1}}{\partial \rho u}\{v\} \\ \frac{\partial \mathscr{F}_1^{\#,1}}{\partial \rho}\{H\} + \frac{\mathscr{F}_1^{\#,1}}{2}(\frac{-\rho e - p}{\rho^2} + \frac{1}{\rho}\frac{\partial p}{\partial \rho}) & \frac{\partial \mathscr{F}_1^{\#,1}}{\partial \rho u}\{H\} + \frac{\mathscr{F}_1^{\#,1}}{2}(\frac{1}{\rho}\frac{\partial p}{\partial \rho u}) \\ 0 & 0 \\ \frac{\partial p}{\partial \rho v} & \frac{\partial p}{\partial \rho E} \\ \frac{\mathscr{F}_1^{\#,1}}{2\rho} & 0 \\ \frac{\mathscr{F}_1^{\#,1}}{2}(\frac{1}{\rho}\frac{\partial p}{\partial \rho v}) & \frac{\mathscr{F}_1^{\#,1}}{2}(\frac{1}{\rho}(1 + \frac{\partial p}{\partial \rho e})) \end{pmatrix}.$$

The derivatives of the fluxes in the y-direction can be obtained in an analogous way.

2.4.2 Aeroacoustics of Flow over a Cavity

Considering the two-dimensional flow over a cavity, different acoustic phenomena can be expected, depending on the cavity shape and the inflow conditions. The generated noise allows to evaluate the properties of the implicit-in-time split DG scheme. We choose $Re_D = 1500$ based on the cavity depth and $Ma = 0.6$ as inflow conditions. The cavity has a ratio of length to depth of $L/D = 2$. Further details on the setup are given in [18].

The influence of the time step size on the obtained results is illustrated with the 4^{th} order 6-stages ESDIRK scheme from [15]. As a comparison, the explicit low storage 4^{th} order 5-stages Runge-Kutta scheme from [8] with a CFL number of CFL $= 0.9$ is used. To relate the time step of the implicit scheme to physics, the given CFL numbers are calculated with the CFL condition of the explicit scheme. This value of the Courant number corresponds to the physical time scale and guarantees the proper time resolution of the acoustic waves. For the spatial discretization a polynomial degree of $N = 5$ is chosen. The total amount of approximately 7.2×10^4 DOFs is distributed on the cores such that the load is ≈ 1000 #DOF/Core. In an explicit simulation we

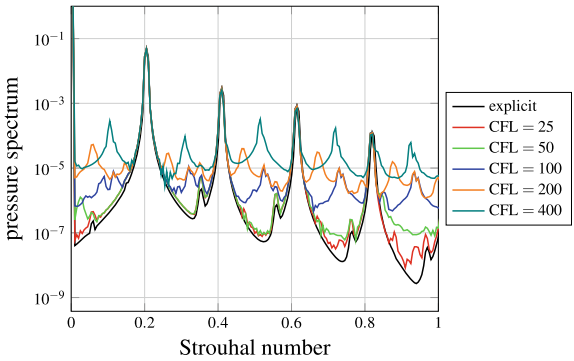


Fig. 1 Pressure spectrum of flow over cavity with different CFL numbers, using the implicit-in-time split DG formulation and a temporal resolved explicit-in-time formulation as a comparison.

Table 1 Relative CPU times with respect to the explicit scheme for simulation of flow over cavity with different time step sizes of implicit scheme.

CFL	400	200	100	50	25
rel. CPU time	0.78	1.02	1.55	2.62	4.63

have to resolve the acoustic wave propagation well – otherwise the scheme becomes unstable. Increasing the Courant number results in an underresolution of the acoustics and the numerical results do no longer show the proper acoustics.

Figure 1 shows the spectrum of the pressure fluctuations for different Courant numbers, i.e. for different time steps. As expected for this configuration, tonal peaks are visible - the so-called Rossiter-modes [22]. These modes are amplified by acoustic feedback. In Fig. 1 it is shown that for all Courant numbers under considerations the frequencies of this tonal noise are visible. However, the spectrum changes by increasing the Courant number and time step size. Additional pressure peaks occur and hide the tonal peaks at higher frequencies. For our time approximation, this test setup shows that the requirement of resolving all temporal phenomena can be relaxed. The implicit time discretization remains stable in the under-resolved situation.

In Table 1 the relative computational times with respect to the explicit scheme are given. It shows that if the considered phenomena allow large time steps, a gain in the required computational time compared to the explicit scheme can be achieved by using implicit time discretization. The resulting efficiency gain becomes higher for lower Mach numbers. This work shows the stability for under-resolved modes and lays the groundwork to extend the range of problems, which can be tackled by FLEXI, into the low Mach number regime.

3 Zonal Large Eddy Simulation of a Compressible Turbulent Boundary Layer

In order to tackle the LES of multiphysics problems, such as the interaction of shocks with turbulent boundary layers (TBL) in combination with fluid structure interaction,

we need an accurate and efficient method to compute TBL. Within this subproject, we perform a zonal, wall-resolved LES (ZLES) of a compressible TBL on a rigid flat plate with zero pressure gradient. The results are validated against DNS data of Wenzel et al. [28]. A combination of the recycling rescaling technique and the anisotropic linear forcing (RRALF) according to Kuhn et al. [19] is chosen as the ZLES ansatz to provide physically consistent inflow data.

3.1 Numerical Setup

The ZLES simulation of the TBL is based on the setup of Wenzel et al. [28]. For our validation, the $Ma = 2.0$ case is chosen at an inflow Reynolds number of $Re_\theta = 1100$ based on the momentum thickness. The thermodynamic properties are given by $T_\infty = 288.15\ K$, $Pr = 0.72$, $\gamma = 1.4$, $R = 287\ J/(kgK)$ and the viscosity is derived from Sutherland's law. The computational domain has the size of $40\delta_{in} \times 10\delta_{in} \times 5\delta_{in}$ with $144 \times 46 \times 30$ grid cells in streamwise, wall-normal and spanwise direction, respectively. At a polynomial degree of $N = 5$, this leads to a resolution at the inlet of $\Delta x^+ = 25$, $\Delta y_w^+ = 1.5$, $\Delta z^+ = 15$ in viscous wall units and about 43 million degrees of freedom in total. The wall is treated as an adiabatic no-slip wall and periodic boundary conditions are used in the spanwise direction. Far field condition is adopted as top boundary. The outlet is handled as supersonic outlet.

The crucial part of ZLES is the turbulent inflow method, here the RRALF method is utilized. In this case the recycling plane is positioned $8\delta_{in}$ downstream of the inlet. Within the forcing zone, the first and second order turbulent statistics are controlled, which guarantees long-term stability of the recycling technique. The data from [28] is used as target statistics. The recycling plane as well as the forcing zone are displayed in Fig. 2. To ensure the flow becomes rapidly turbulent, the flat plate is initialized with the reference data superimposed with wall-parallel streaks.

3.2 Results

The simulation was performed with 2560 CPU cores, which leads to a load of about 17000 #DOF/Core. The Performance Index (PID) of this simulation is at about $1.7\,\mu s$. The PID is defined as

$$\text{PID} = \frac{\text{wall-clock-time} \cdot \#\text{cores}}{\#\text{DOF} \cdot \#\text{time steps} \cdot \#\text{RK-stages}}\ , \tag{6}$$

and describes the time it takes one core to update one DOF for one time stage of the explicit Runge-Kutta method. Here the PID is slightly larger compared to the standard implementation of FLEXI. This is due to the overhead introduced by the RRALF method. The latest code optimizations within the RRALF method reduce

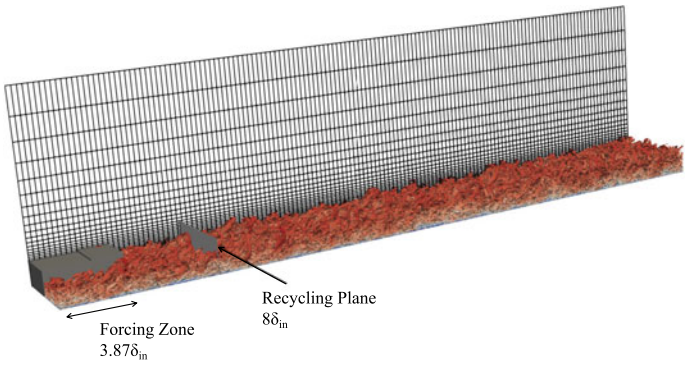


Fig. 2 Computational domain of the flat plate simulation. Q-criterion visualization of turbulent structures colored by velocity magnitude. In the inflow region the forcing zone and the recycling plane are shown half in spanwise direction.

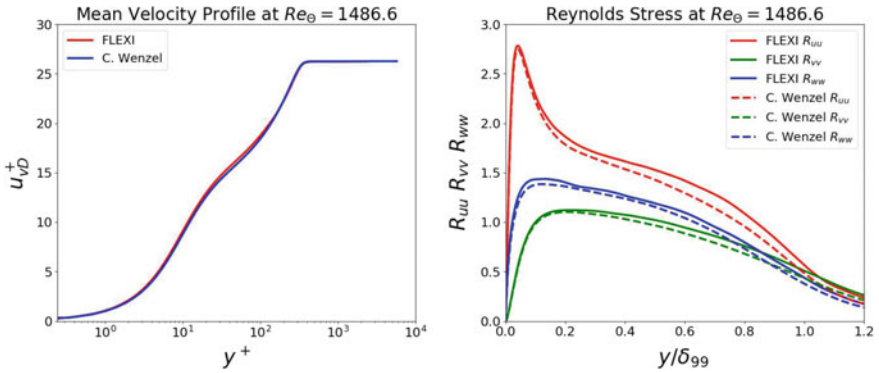


Fig. 3 Van Driest transformed velocity profiles (left) and Morkovin's density scaled Reynolds stress profiles (right).

the PID by approximately 15%. The slightly increased PID value is justified by the high reduction in computing time due to the ZLES approach.

The flow reached a quasi-steady state after $t = 7t_f$, where t_f is the domain flow through time. Figure 2 shows the instantaneous isosurfaces of the Q-criterion, colored by velocity magnitude at the quasi-steady state. From here on the turbulence statistics are collected for further $4.4t_f$. Figure 3 compares the van Driest transformed mean velocity profiles and the density-scaled Reynolds stresses according to Morkovin's density scaling law with DNS data provided by Wenzel et al. [28] at $x = 20.5\delta_{in}$ ($Re_\theta = 1486.6$). The scaling laws are:

$$u^+_{vD} = \int_0^{u^+} \sqrt{\rho/\rho_w}\, du^+ \quad \text{and} \quad R_{\phi\phi} = \sqrt{\phi'\phi'} \cdot \frac{\sqrt{\rho/\rho_w}}{u_\tau}.$$

The mean velocity profiles of the ZLES are in good agreement with the DNS data with slight deviations at the boundary layer edge. Also the Reynolds stresses show good agreement with the DNS data. The RRALF method slightly overpredicts the Reynolds stresses, especially R_{uu} towards the edge of the boundary layer.

Besides the mean velocity and Reynolds stress profiles, thermodynamic fluctuations, e.g. density, pressure and temperature fluctuations, of the flow will be studied in the future, since they are especially significant for the analysis of compressibility effects [21]. This simulation serves as a starting point for complex fluid-structure interactions triggered by shock/boundary layer interactions on compliant walls.

4 Large Eddy Simulation of Multi-stage Turbomachinery Using a Sliding Mesh Method

Modern aircraft engines aim to provide increased performance with a decreasing number of blades and stages. Both trends result in smaller and lighter engines, thereby increasing fuel efficiency [12]. The application of ultra high lift aerofoils in the low-pressure turbine is of crucial importance to achieve these goals. The aerofoils rely on unsteady wake-boundary layer interaction to improve the performance characteristics beyond the capabilities of stationary flows. For these applications, multi-stage simulations are crucial in order to accurately predict the flow field. However, the capturing of unsteady effects in the boundary layer region poses a significant challenge for numerical simulations. The traditional approach using Reynolds-Averaged Navier-Stokes (RANS) modeling and its unsteady extension (URANS) keeps the computational effort low but fails to provide time-accurate data and relies on turbulence models including their known shortcomings to approximate the crucial boundary layer region [24]. Large Eddy Simulation (LES) alleviates many of those problems at the cost of significantly increased computational requirements. Simulations are often performed with a wall-modeled approach, carrying some of the model dependence of RANS over to LES [7]. Only recently, wall-resolved LES of multi-stage setups became feasible. In this context, we present the wall-resolved LES of a subsonic multi-stage turbine cascade utilizing a sliding mesh approach.

4.1 Implementation of the Sliding Mesh

The DG-solver FLEXI was extended by a sliding mesh method proposed by Zhang and Liang [29] to account for the relative movement of the blade stages. The sliding mesh method allows subdomains to slide along common interfaces, as indicated in Fig. 4. While the mesh movement itself can be incorporated with the common arbitrary Lagrangian-Eulerian (ALE) approach, e.g. [14], the coupling of the subdomains along the common interface emerges as crucial for retaining the high-order accuracy

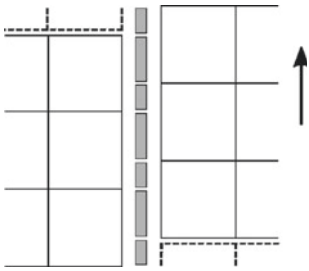


Fig. 4 Schematic view of the sliding mesh interface with periodic boundary conditions. The gap between the subdomains reveals the mortars in grey.

of the underlying numerical method as well as the solver's scalability on massively parallel systems.

The DG method uses local ansatz functions in each individual element with the elements only coupled weakly by the numerical fluxes at the elements' faces. Hence no volumetric data has to be exchanged for the coupling at the sliding mesh interfaces. The implemented method introduces two-dimensional *mortars* at the interface as shown in Fig. 4. In each time step the local solution at the element faces is projected onto the corresponding mortars, where the numerical flux function is evaluated and subsequently projected back onto the element faces on both sides of the interface. This retains the high-order accuracy of the method provided that L_2-projections of the respective order are used [29].

The key challenge in terms of parallelization of the method is to retrieve the constantly shifting communication partners at the interface during simulation. By enforcing an equidistantly spaced mesh at the interface, the adjacent elements and their respective ranks can be recovered analytically at every time instant as long as the velocity of the mesh movement is known. This can in turn be exploited to build a local communication scheme in which each processor can deduce all processors which require information exchange without the need for additional communication during the simulation. The only additional global communication occurs during preprocessing and initialization routines.

4.2 Performance

To quantify the introduced overhead and to assess the suitability of the proposed implementation for large-scale applications, the scaling of the code on the Cray XC40 system Hazel Hen was investigated. As test case a cubical Cartesian mesh as shown in Fig. 5 was initialized with a constant freestream and constant mesh velocity for the center subdomain yielding two sliding mesh interfaces. The investigated number of cores ranged from 24 cores (1 node) to a maximum of 12,288 cores (512 nodes). Starting from a baseline mesh with 6^3 elements, the number of elements in the mesh was successively doubled in the three spatial dimensions until the minimum of 9 elements per core is reached for the respective number of processors used. The

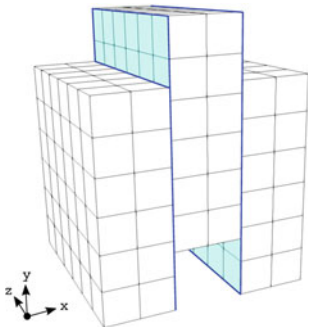


Fig. 5 Baseline mesh with 6^3 elements for the scaling test. The middle subdomain is moving along two planar sliding mesh interfaces highlighted in blue.

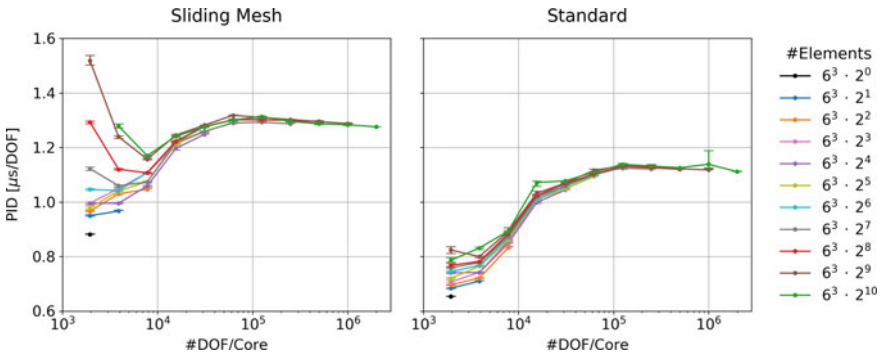


Fig. 6 Results of the scaling test. Shown is the PID for several amounts of processor cores and mesh sizes. For every data point the mean of five runs is given with the maximum and minimum given as errorbars. The results of the sliding mesh implementation are shown on the left and the results of the baseline code on the right.

polynomial degree was set to five, resulting in a sixth-order scheme. The quantity of interest is the Performance Index (PID) as defined in Eq. (6).

Each run was carried out five times with the results in Fig. 6 showing the mean, minimum and maximum for each data point. The results for the baseline code are also given for comparison. For loads of more than 10,000 #DOF/Core, the qualitative behaviour of the sliding mesh implementation matches the scaling behaviour of the baseline code with a decrease in performance by about 20%. This is caused by the additional work for the moving mesh and the overhead of the sliding mesh implementation. For small loads of less than $5,000$ #DOF/Core, the performance degenerates substantially with an increasing amount of computing cores. This is due to the additional communication and the load imbalance at the interface which cannot be hidden effectively for small loads. Since loads of $10,000$ #DOF/Core and more are reasonable for many large-scale applications, the implementation proves itself suitable for HPC.

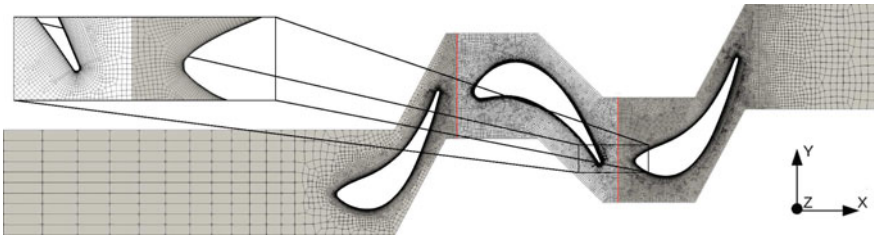


Fig. 7 Cross section of the computational grid. The sliding mesh interfaces are highlighted in red.

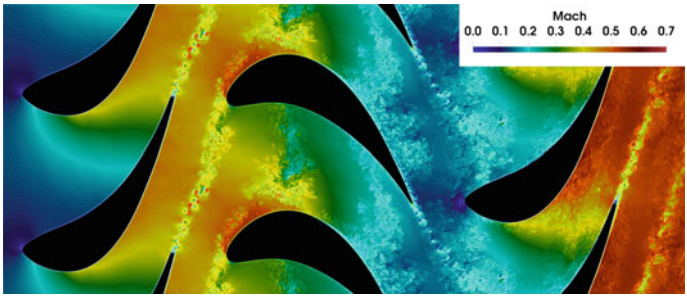


Fig. 8 Snapshot of the instantaneous Mach number distribution.

4.3 Test Case

As test case for turbomachinery application, the mean line geometry of the Aachen 1.5 axial flow turbine featuring a stator-rotor-stator configuration was chosen. The stator profiles are geometrically identical Traupel profiles while the rotor uses a modified VKI geometry. The original blade count of 36-41-36 is changed to a 38-38-38 configuration of which one pitch was simulated. The respective inlet and outlet Mach number are $Ma_{inlet} = 0.1$ and $Ma_{outlet} = 0.5$, resulting in a chord Reynolds number $Re = \frac{U_\infty c}{\nu} = 8 \times 10^5$ based on the stator chord [10, 26, 27].

A cross section of the computational domain with a zoom on the boundary layer refinement is shown in Fig. 7. The mesh consists of an unstructured grid in the freestream and a structured O-type mesh in the blade vicinity for a total of 966288 cells. Red lines denote the positions of the sliding mesh interfaces using the approach outlined in Sect. 4.1. The polynomial degree of the implicit LES is set to sixth order resulting in 2.087×10^8 degrees of freedom (DOF). A total of 18 900 record points are placed in the boundary layer region which capture the time-accurate solution at every iteration. The information in spectral space is obtained by an FFT. The simulation is ramped up until a periodic state is achieved and subsequently sampled for 10 blade passing periods.

The Mach number of the instantaneous fluid solution at the end of the simulation is shown in Fig. 8. At this instance, the rotor encounters unsteadiness exclusively caused by the trailing edge vortices of the upstream stator. The wakes remain recognizable

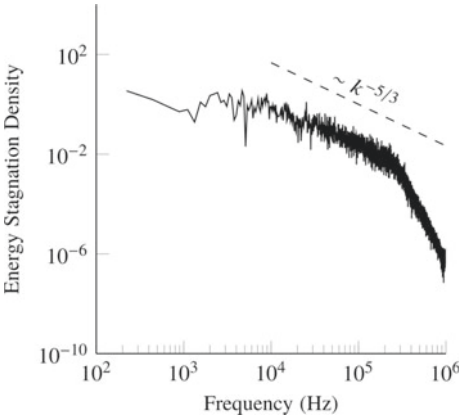


Fig. 9 Power spectral density of turbulent kinetic energy near the trailing edge of the rotor suction side.

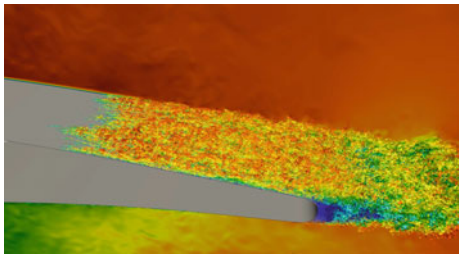


Fig. 10 Instantaneous fluid solution around the trailing edge of second stator using iso-surfaces of the Q-criterion.

as they are bent throughout the rotor passage and are separated by regions of calmed fluid. As these regions persist until the downstream vane row, the flow at second stator's leading edge also experiences distinct changes of turbulent intensity over time.

The spatial and temporal resolution allows for the analysis of turbulence development and non-linear interactions. One such example is the power spectral density plot of turbulent kinetic energy at 90 % rotor surface length in Fig. 9. The energy contained in wave numbers associated with the blade passing frequency at 2216.66 Hz is clearly recognizable. The extensive inertial subregion indicates a well developed turbulent spectrum at this probing location which was chosen to emphasize the predictive accuracy of the simulation.

However, significantly more important for performance assessment of the turbine case are regions such as the aft section of the second stator as shown by isosurfaces of the Q-criterion in Fig. 10. As the change from a laminar to a turbulent boundary layer state significantly influences skin friction and heat transfer, the exact location of the transition point is of the essence. Such examples make another strong point on spending the computational effort on wall-resolved calculations rather than relying on wall modeling.

5 Summary and Outlook

In this report, we have presented the current capabilities of FLEXI with regards to complex large scale simulations as well as new numerical developments. The inclusion of flexible temporal integration schemes allows us to considerable broaden the scope of applicability of FLEXI, for e.g. RANS formulations, low Mach number flows or other stiff problems. The LES computation of the turbulent compressible boundary as well as the rotor-stator-interactions presented here show how large scale simulations with FLEXI can be used to predict flow physics in complex situations and investigate strongly coupled non-linear interactions.

Recently, FLEXI has also been successfully ported to the new HAWK architecture. The necessary changes to the code and its build environment induced by the switch to HAWK have been minor. Preliminary scaling results on the system in the test phase indicate that the scaling of FLEXI remains stable, however, further work is required on the investigation of the influence of the reduced memory bandwidth on node level by the means of striding. Furthermore, the change from Cray Aries to the hypercube InfiniBand network will be investigated. This will be tackled once the system is in production phase.

Acknowledgements We thank the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) for supporting this work by funding - EXC2075 – 390740016 under Germany's Excellence Strategy. We acknowledge the support by the Stuttgart Center for Simulation Science (SimTech) and the DFG International Research Training Group GRK 2160. Min Gao recognizes the support of the China Scholarship Council (CSC). We all truly appreciate the ongoing kind support by HLRS in Stuttgart.

References

1. M. Atak, A. Beck, T. Bolemann, D. Flad, H. Frank, F. Hindenlang, C.-D. Munz, Discontinuous Galerkin for high performance computational fluid dynamics, in *High Performance Computing in Science and Engineering 2014* (Springer International Publishing, 2015), pp. 499–518
2. A. Beck, T. Bolemann, D. Flad, H. Frank, N. Krais, K. Kukuschkin, M. Sonntag, C.-D. Munz, Application and development of the high order discontinuous Galerkin spectral element method for compressible multiscale flows, in *High Performance Computing in Science and Engineering'17* (Springer, 2018), pp. 387–407
3. A. Beck, P. Ortwein, P. Kopper, N. Krais, D. Kempf, C. Koch, Towards high-fidelity erosion prediction: on time-accurate particle tracking in turbomachinery. Int. J. Heat Fluid Flow **79**, 108457 (2019)
4. A.D. Beck, T. Bolemann, D. Flad, H. Frank, G.J. Gassner, F. Hindenlang, C.-D. Munz, High-order discontinuous Galerkin spectral element methods for transitional and turbulent flow simulations. Int. J. Numer. Meth. Fluids **76**(8), 522–548 (2014)
5. A.D. Beck, T. Bolemann, D. Flad, N. Krais, J. Zeifang, C.-D. Munz, Application and development of the high order discontinuous Galerkin spectral element method for compressible multiscale flows, in *High Performance Computing in Science and Engineering'18* (Springer, 2019), pp. 291–307
6. A.D. Beck, G.J. Gassner, T. Bolemann, H. Frank, F. Hindenlang, C.-D. Munz, Underresolved turbulence simulations with stabilized high order discontinuous Galerkin methods, in *Direct and Large-Eddy Simulation IX* (Springer, 2015), pp. 103–108

7. S. Bocquet, P. Sagaut, J. Jouhaud, A compressible wall model for large-eddy simulation with application to prediction of aerothermal quantities. Phys. Fluids **24**(6), 065103 (2012)
8. M.H. Carpenter, C.A. Kennedy, Fourth-order 2N-storage Runge-Kutta schemes, NASA TM-109112 (National Aeronautics and Space Administration, Langley Research Center, Hampton, VA, 1994)
9. H.M. Frank, C.-D. Munz, Large eddy simulation of tonal noise at a side-view mirror using a high order discontinuous Galerkin method, in *22nd AIAA/CEAS Aeroacoustics Conference*, pp. 2847 (2016)
10. H. Gallus, Ercoftac test case 6: axial flow turbine stage, in *Seminar and Workshop on 3D Turbomachinery Flow Prediction III, Les Arcs, France* (1995)
11. G.J. Gassner, A.R. Winters, D.A. Kopriva, Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations. J. Comput. Phys. **327**, 39–66 (2016)
12. F. Haselbach, H.-P. Schiffer, M. Horsman, S. Dressen, N. Harvey, S. Read, The application of ultra high lift blading in the BR715 LP turbine. J. Turbomach. **124**(1), 45–51 (2001)
13. F. Hindenlang, G. Gassner, C. Altmann, A. Beck, M. Staudenmaier, C.-D. Munz, Explicit discontinuous Galerkin methods for unsteady problems. Comput. Fluids **61**, 86–93 (2012)
14. C.W. Hirt, A.A. Amsden, J. Cook, An arbitrary Lagrangian-Eulerian computing method for all flow speeds. J. Comput. Phys. **14**(3), 227–253 (1974)
15. C.A. Kennedy, M.H. Carpenter, Additive Runge-Kutta schemes for convection-diffusion-reaction equations. Appl. Numer. Math. **44**, 139–181 (2003)
16. D.A. Knoll, D.E. Keyes, Jacobian-free Newton-Krylov methods: a survey of approaches and applications. J. Comput. Phys. **193**, 357–397 (2004)
17. N. Krais, A. Beck, T. Bolemann, H. Frank, D. Flad, G. Gassner, F. Hindenlang, M. Hoffmann, T. Kuhn, M. Sonntag et al., Flexi: a high order discontinuous Galerkin framework for hyperbolic-parabolic conservation laws. Comput. Math. Appl. **81**, 186–219 (2020)
18. T. Kuhn, J. Dürrwächter, F. Meyer, A. Beck, C. Rohde, C.-D. Munz, Uncertainty quantification for direct aeroacoustic simulations of cavity flows. J. Theoret. Comput. Acoust. **27**(01), 1850044 (2019)
19. T. Kuhn, D. Kempf, A. Beck, C.-D. Munz, A novel turbulent inflow method for zonal large eddy simulations with a discontinuous Galerkin solver
20. S. Pirozzoli, Numerical methods for high-speed flows. Annu. Rev. Fluid Mech. **43**, 163–194 (2011)
21. S. Pirozzoli, M. Bernardini, Turbulence in supersonic boundary layers at moderate Reynolds number. J. Fluid Mech. **688**, 120–168 (2011)
22. J. Rossiter, Wind tunnel experiments on the flow over rectangular cavities at subsonic and transonic speeds, Technical report, Ministry of Aviation; Royal Aircraft Establishment; RAE Farnborough, 1964
23. Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving non-symmetric linear systems. SIAM J. Sci. Stat. Comput. **7**, 856–869 (1986)
24. J.C. Tyacke, P.G. Tucker, Future use of large eddy simulation in aero-engines. J. Turbomach. **137**(8), 081005 (2015)
25. S. Vangelatos, On the efficiency of implicit discontinuous Galerkin spectral element methods for the unsteady compressible Navier-Stokes equations, Ph.D. thesis, University of Stuttgart, 2019
26. T. Volmar, B. Brouillet, H. Benetschik, H. Gallus, Test case 6: 1-1/2 stage axial flow turbine-unsteady computation, in *ERCOFTAC Turbomachinery Seminar and Workshop* (1998)
27. R. Walraevens, H. Gallus, Testcase 6-1-1/2 stage axial flow turbine. Ercoftac Testcase **6**, 201–212 (1997)
28. C. Wenzel, B. Selent, M. Kloker, U. Rist, DNS of compressible turbulent boundary layers and assessment of data/scaling-law quality. J. Fluid Mech. **842**, 428–468 (2018)
29. B. Zhang, C. Liang, A simple, efficient, and high-order accurate curved sliding-mesh interface approach to spectral difference method on coupled rotating and stationary domains. J. Comput. Phys. **295**, 147–160 (2015)

Numerical Analysis of Stratified T-Junction Mixing Flows

Cenk Evrim and Eckart Laurien

Abstract Mixing of coolant streams at high temperature differences in a T-junction frequently cause unexpected fatigue cracking of piping material, this is a challenge for the safe operation of a nuclear power plant. This study compares the T-junction flow mixing in different T-junction configurations of the Fluid-Structure Interaction (FSI) facility at the University of Stuttgart. Warm and cold fluids flow in the main and branch pipes with a temperature difference $\Delta T = 180\,K$ between the mixing fluids. Numerical study is conducted using the Large-Eddy Simulation (LES) method. The numerical results are validated with in-house experimental data from the horizontal and vertical T-junction configuration. The stratified T-junction mixing flows are numerically investigated and characterized. Additionally, this paper also contains a close investigation of temperature fluctuations and the influence of the weld seam on the mixing process.

1 Introduction

A nuclear power plant contains a vast piping network supply coolant to the required locations in order to ensure its safe operation. There are certain locations where the fluids at significant temperature differences are mixed to control the resulting outflow temperature. Such mixing occurs mainly in T-junction piping configurations located in the residual heat removal system, emergency core cooling system and charging lines. The mixing results in random thermal fluctuations of varying amplitudes over a wide range of frequencies to be imposed on the piping wall. Depending on the

C. Evrim (✉) · E. Laurien
Institute of Nuclear Technology and Energy Systems, University of Stuttgart,
Pfaffenwaldring 31, 70569 Stuttgart, Germany
e-mail: cenk.evrim@ike.uni-stuttgart.de

E. Laurien
e-mail: eckart.laurien@ike.uni-stuttgart.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_23

pipe geometry, material properties and thermal loading on the structure crack initiation could be expected during any timeframe resulting in high-cycle thermal fatigue (HCTF) damage of piping components. Since the thermal loading caused by such mixing is of lower amplitude, monitoring such loads using conventional surface thermocouple instrumentation is found to be highly ineffective as the underlying flow is highly turbulent and exhibits complex mixing behavior. A well-known example is the cracking incident reported by Chapuliot et al. [1] in the Civaux-1 NPP during 1998 where through-wall crack over a length of 180 mm and a network of other cracks developed within a short span of 1500 h near the T-junction piping in the RHRS. Metallurgical investigations concluded the origin of the cracking was caused by thermal fatigue. Following this incident, T-junction flow mixing using different piping material, inflow velocities and temperatures were experimentally investigated in literature. A list of such experiments is provided in Selvam et al. [12].

Aside from measurements, advances in high-speed computing have propelled the field of computational fluid dynamics (CFD) to the forefront of both the academia and industry over the past two decades. CFD studies of nuclear safety related research has been widely carried out by various investigators employing different methodologies and a description of the research areas is provided in Smith et al. [13]. The OECD/NEA organized a CFD benchmarking exercise based on the T-junction measurements carried out at the Vattenfall test facility in Sweden [13]. The majority of participants in this exercise used the Large-Eddy Simulation (LES) method, e.g. Höhne [5] and Jayaraju et al. [6], which offered a good trade-off between computational expense and accuracy of solutions in comparison with the Reynolds Averaged Navier-Stokes (RANS) method. In this investigations are rather coarse meshes used due to high Reynolds numbers.

To determine the conjugate heat transfer and to predict the characteristics of near-wall turbulence, the thermal fluid-structure interaction and heat conduction of the solid material must be taken into account. Generally two basic approaches can be utilized: (i) resolving the near-wall layer or (ii) by wall functions, to bridge the gap between the wall and the first node. The approach (i) is called a wall-resolved LES.

Numerical simulations based on wall functions tend to underestimate the flow and thermal fluctuations which effect the heat transfer through the walls [6]. It is therefore suggested to resolve the near-wall layer, then the flow structures close to the wall are directly well resolved.

If the temperature difference is small, e.g. [13], the buoyancy effect may be neglected. The gravity and buoyancy effect is significant at higher temperature differences because this leads to a turbulence damping with a stabilization in addition to a wavy density stratification of the mixing fluids. To investigate turbulent mixing mechanisms with high temperature differences and buoyancy effects the Fluid-Structure Interaction (FSI) facility at the University of Stuttgart was designed and built, see Kuschewski et al. [7].

In earlier numerical investigations with the LES approach, a wavy stable stratification could be detected in a horizontal T-junction configuration, e.g. by [12]. The highest thermal fluctuations are located on the interface between warm and cold areas. Indeed, the mixing mechanisms in vertical T-junction configurations and the

effect of the weld seam is still not clear, in view of the fact that no numerical studies have been performed. The aim of this research paper is to fill the knowledge gap by supplying wall-resolved LES of stratified mixing flows both in the horizontal and vertical T-junction configuration. To achieve the target, we analyze the influence of the inflowing cold branch pipe fluid on the mixing process. The predicted LES results are validated with experimental data of the FSI facility. Additionally, to analyze the buoyancy effect, the results are compared with each other. Furthermore, this paper includes the impact of the weld seam on the heat transfer and mixing process.

2 Computational Details

2.1 Governing Equations

In numerical flow simulations with LES method, the three dimensional large-scale structures are directly resolved, while the small-scale structures in the turbulent flow are modeled. In turbulent flows of variable density fluids, the tilde (∼) denotes Favre-filtered (density-weighted) variables, while the overbar (-) presents Reynolds-filtered variables. In the following Eq. 1–3 represent the mass, momentum and energy conservation for the current LES, respectively:

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial \bar{\rho}\tilde{u}_j}{\partial x_j} = 0 \tag{1}$$

$$\frac{\partial \bar{\rho}\tilde{u}_i}{\partial t} + \frac{\partial \bar{\rho}\tilde{u}_i\tilde{u}_j}{\partial x_j} = -\frac{\partial \bar{p}}{\partial x_j} + \frac{\partial}{\partial x_j}\left[\left(\frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i}\right) - \frac{2}{3}\frac{\partial \tilde{u}_k}{\partial x_k}\delta_{ij}\right]\mu_{eff} + \bar{\rho}g_i \tag{2}$$

$$\frac{\partial \bar{\rho}\tilde{h}}{\partial t} + \frac{\partial \bar{\rho}\tilde{u}_j\tilde{h}}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\alpha_{eff}\frac{\partial \tilde{h}}{\partial x_j}\right) \tag{3}$$

The effective dynamic viscosity is the sum of the molecular and turbulent viscosities and the effective diffusivity is consisting of the molecular and turbulent diffusivities. The turbulent Prandtl number Pr_t has the value 0.85. The turbulent eddy viscosity is calculated by using the Wall Adapting Local Eddy (WALE) subgrid-model [9] and is expressed by

$$\mu_t = \bar{\rho}\nu_t = \bar{\rho}\left(C_w\Delta\right)^2 \frac{\left(S_{ij}^d S_{ij}^d\right)^{3/2}}{\left(\tilde{S}_{ij}\tilde{S}_{ij}\right)^{5/2} + \left(S_{ij}^d S_{ij}^d\right)^{5/4}} \tag{4}$$

here $\tilde{S}_{ij}$ defines the resolved shear strain rate tensor and S_{ij}^{d} represents the traceless symmetric part of the square of the velocity gradient tensor and is expressed by $S_{ij}^{d} = \frac{1}{2}\left(\tilde{g}_{ij}^{2} + \tilde{g}_{ji}^{2}\right) - \frac{1}{3}\left(\delta_{ij}\tilde{g}_{kk}^{2}\right)$ with the velocity gradient tensor $\tilde{g}_{ij} = \frac{\partial \tilde{u}_i}{\partial x_j}$. The WALE constant C_w has the value 0.325.

Thermal conductivity inside the solid region is considered by the thermal diffusion equation whose formulation results from Eq. 3 by neglecting the convection term:

$$\frac{\partial\left(\rho_s h_s\right)}{\partial t} = \frac{\partial}{\partial x_j}\left(\alpha_s \frac{\partial h_s}{\partial x_j}\right) \tag{5}$$

Where α_s represents the thermal diffusivity of the piping material. The wall temperature T_w is estimated by coupling the local heat fluxes of the solid and fluid domain by using a linear approximation:

$$\frac{\lambda\left(\tilde{T}_1\right)}{\Delta y_1}\left(\tilde{T}_1 - T_w\right) = \frac{\lambda\left(T_s\right)}{\Delta y_{1,s}}\left(T_w - T_s\right) \tag{6}$$

here Δy denotes the wall distance, while the indices defines the nearest cell centers from the wall in the fluid and solid part, respectively.

2.2 Computational Domain and Numerical Method

The computational domain and the boundary conditions are based on the FSI experiment in Zhou et al. [14]. An illustration of the computational domain of both configurations and a view of the weld seam geometry is shown in Fig. 1(a–f). The simulation domain consists of an inner main pipe diameter of 0.0718 m (D_m) and an inner branch pipe diameter of 0.0389 m (D_b). It has a total downstream length of 12 diameters ($12D_m$). The T-junction is discretized with a total of 17.5 Mio. (fluid region: 12.2 Mio.; solid region: 5.3 Mio.) structured hexahedral mesh.

The numerical mesh is based on the best practice guidelines by Piomelli and Chasnov [10] for a wall-resolved LES. The non-dimensional length, height and width of the first cell close to the wall measured in wall units are defined as Δx^+, Δy^+ and Δz^+, respectively. They assess the resolution of the mesh. The superscript "+" represents a dimensionless length normalized by the friction velocity u_τ and the kinematic viscosity ν. The first cell heights within the boundary layer is $6*10^{-6}$ m, which corresponds to $\Delta y_{max}^{+} < 1$ throughout the computational domain in the present LES calculations. Table 1 summarizes a comparison of the LES with the recommended range by [10].

To speed up the turbulence development, a perturbation approach by [11] was used. The inflow generator has an initial length of $4D_{m/b}$ and the aim is to generate and provide a fully developed turbulent flow, see e.g. [2, 3, 8]. At first, steady simulations

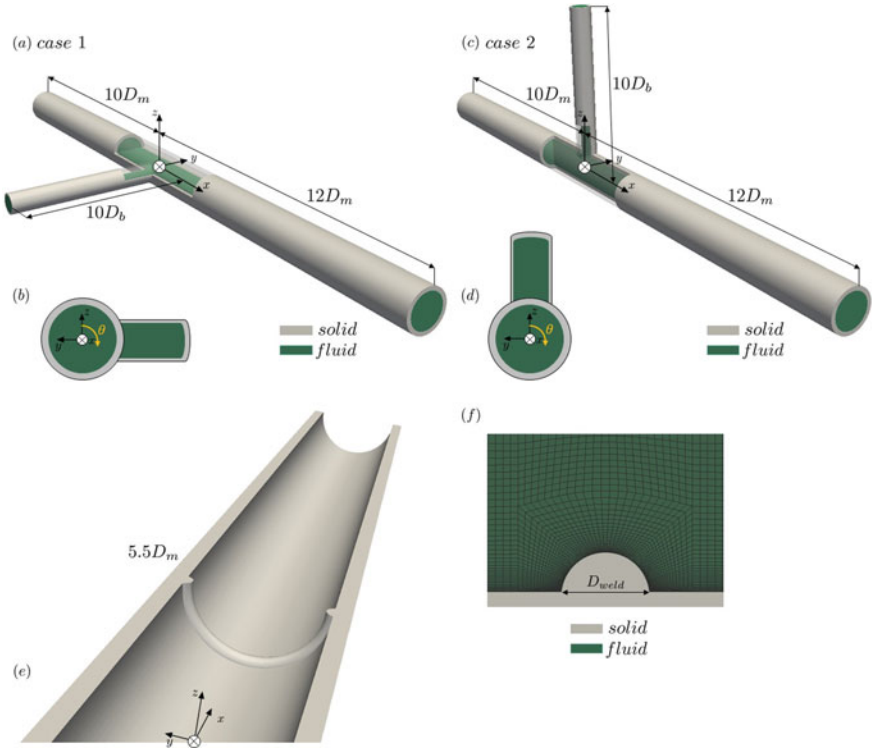


Fig. 1 Flow domain of both configurations and a closer view of the weld seam area

Table 1 Comparison of the mesh resolution of the current LES with recommended values in the literature

	Δx^+	Δy^+	Δz^+
[10]	50–150	1	15–40
Present	<67	<1	<19

are performed based on k-ω-SST turbulence model. The temperature distribution of the steady solution is used for the initialization of the LES.

The temperature difference between the mixing fluids is ΔT = 180 K with a mass flow rate ratio ($\dot{m}_m/\dot{m}_b$) of 3. Fixed temperature inlet values are used at each flow inlet. The numerical simulations are coupled with the heat conduction in the solid material in order to take account of the thermal fluid-structure interaction. Outer walls and the outlet are set as an adiabatic boundary condition. A non-slip boundary condition for the velocity is selected on the inner wall. Fluid properties are temperature dependent and are specified as polynomials based on the NIST database. Constant thermophysical properties are considered for the solid region: $\rho_s = 8000$ kg/m^3, $c_{p,s} = 500$ J/(kgK) and $\lambda_s = 15$ W/(mK). Table 2 contains the inflow parameters of the investigated flow mixing process.

Table 2 Simulation conditions

p [bar]	Re_m [−]	Re_b [−]	$\dot{m}_m$ $[kg/s]$	$\dot{m}_b$ $[kg/s]$	T_m $[K]$	T_b $[K]$	Pr_m [−]	Pr_b [−]
75	78400	6600	0.6	0.2	473.15	293.15	0.9	7

The aforementioned governing equations are discretized with the open-source finite-volume code OpenFOAM v5.0. The pressure and velocity term in the momentum equation are coupled by using the PIMPLE algorithm. Different discretization schemes were applied for solving PDEs; spatial discretization was made with the central differencing scheme and temporal discretization was performed by a second order implicit backward time discretization scheme. Convective terms in the energy equation were solved by the second order scheme limitedLinear and fixed blended Linear-Upwind Stabilized Transport (LUST) scheme. A cellMDLimited Gauss linear scheme is applied for the temperature gradient. An unbounded second order scheme is used for the Laplacian terms. For the surface normal gradient terms is an explicit non-orthogonal correction used. The method has overall second-order accuracy in both space and time. The transient simulations are solved with a fixed Courant number of $Co = 1$ and a variable time step. The flow statistics were averaged for at least 30 flow-through times.

3 Results and Discussion

Large-Eddy Simulations have been performed with the aforementioned boundary conditions (see Table 2) and the results are discussed in this section. In this section, the normalized mean temperature $\overline{T^*}$ and the normalized temperature fluctuations T^*_{RMS} are expressed by

$$T^* = \frac{T - T_b}{T_m - T_b} \qquad \overline{T^*} = \frac{1}{N}\sum_{N=1}^{N} T^* \qquad T^*_{RMS} = \sqrt{\frac{1}{N}\sum_{N=1}^{N}\left(T^* - \overline{T^*}\right)^2} \tag{7}$$

here N defines the number of sampling points.

3.1 Validation

Figure 2 shows the validation of the normalized temperature and normalized temperature fluctuations at the position $x = 6.5D_m$ in the near wall region (1 mm into the fluid from the inner wall) of case 1 and 2.

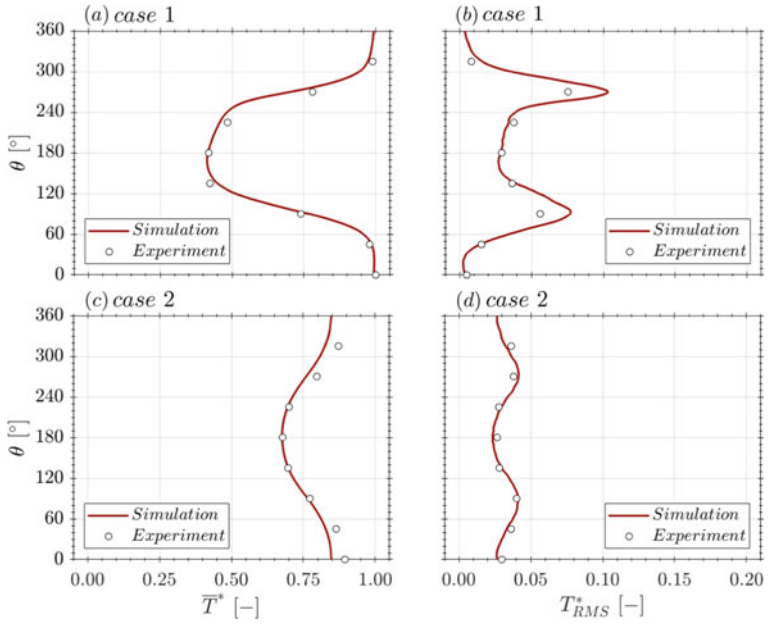


Fig. 2 Comparison of normalized mean temperature **a** case 1 and **c** case 2 and normalized temperature fluctuations **b** case 1 and **d** case 2 at $x = 6.5D_m$

In comparison with the experiment, we infer that our LES results agree well with the measurement data of both cases. This agreement between the experimental and LES results indicates that the current LES results are reasonably reliable for further investigation. In Fig. 2(a), the thermal mixing in case 1 is incomplete with areas which are remaining warm with $\overline{T^*} = 1$. Moreover, the lowest normalized temperature in case 1 is around $\overline{T^*} = 0.4$. The corresponding fluctuations profile in Fig. 2(b) shows a double-peak structure. This indicates that the highest fluctuations are on the interaction layer between warm and cold. The peaks are located at (i) $\theta = 270^o$ with a peak value of 0.105 and (ii) $\theta = 100^o$ with a peak value of 0.075. We note an enhanced thermal mixing in case 2. The fluctuations profile of case 2 shows also a double-peak structure as shown in Fig. 2(d). However, it can be seen that the peak values in case 2 are much smaller than in case 1. Additionally, both peaks in case 2 have the same value of 0.04. This plots shows that the edges of the stratification oscillates strongly which leads to a double-peak formation. However, the core of the stratification oscillates weakly therefore the stratification has a damping behavior, similar to the oberservations in e.g. [4].

3.2 Near-Wall Flow Characteristics

Figure 3(a–d) show the mean normalized temperature $\overline{T^*}$ on circumferential distribution in the near wall region (1 mm from the inner wall into the fluid) with/without weld seam for both cases.

The temperature field in case 1 shows a wavy stable stratification as shown in Fig. 3(a, b). This flow pattern type presents that in the mixing zone are areas where mainly the warm fluid with $\overline{T^*} = 1$ is flowing. We notice cold areas with the value $\overline{T^*} = 0$ in the mixing zone at $0.2D_m \leq x \leq 1.2D_m$. The heavier cold fluid from the branch pipe enters the mixing zone and immediately falls to the bottom and flows in the underpart. Therefore, it allows the lighter warm fluid to stream in the upper part of the pipe. This cold fluid is heated and disappears in the further downstream and the lowest normalized temperature has the value $\overline{T^*} \approx 0.5$. Based on the temperature field close to the weld seam in case 1, we infer that the stratification layer is shifted upwards and downwards due to the presence of the weld seam as shown in Fig. 3(a, b). It can be seen here that Fig. 3(b) represents an undisturbed thermal field.

First of all, we can detect in case 2 an enhanced thermal mixing (see Fig. 3(c, d)). In the near-wall region exist no area with the value $\overline{T^*} = 0$. The thermal field shows here also a stable stratification, but without an oscillating behavior and wavy flow pattern type as in case 1. The cold fluid flows from above into the mixing zone and this observation indicates that the effect of buoyancy is in case 2 stronger and this leads to an improved mixing process. In this configuration can not be seen a similar effect of rotation and shifting of the thermal field in the near-wall region like in case 1.

Figure 4(a–d) show the normalized temperature fluctuations T^*_{RMS} on circumferential distribution in the near wall region (1 mm from the inner wall into the fluid) with/without weld seam for both cases.

The temperature fluctuations field in case 1 shows high intensities in the interaction layer between warm and cold. It can be seen that the peak intensities are existing in the zone where the cold branch pipe fluid enters the warm main pipe. This can be obvserved in the area $126° \leq \theta \leq 180°$ and $0D_m \leq \theta \leq 0.2D_m$ as shown in Fig. 4(a, b). Furthermore, we remark that the peak thermal fluctuation intensities are falling in the further downstream. Due to the presence of the weld seam, we detect that the fluctuation field is also disturbed and shifted. In addition, just shortly after the weld seam the fluctuation values in the near-wall region decrease suddenly (see Fig. 4(a, b)).

In case 2, we can detect areas with high fluctuations intensities at $9° \leq \theta \leq 45°/315° \leq \theta \leq 351°$ and $0D_m \leq \theta \leq 0.6D_m$, see Fig. 4(c, d). This areas appear due to the oscillating impinging jet from the branch pipe. Due to the enhanced thermal mixing, the fluctuations intensities in the further downstream are relatively lower compared to case 1. Furthermore, we note that the weld seam leads to a reduction of the thermal fluctuations in the near-wall region.

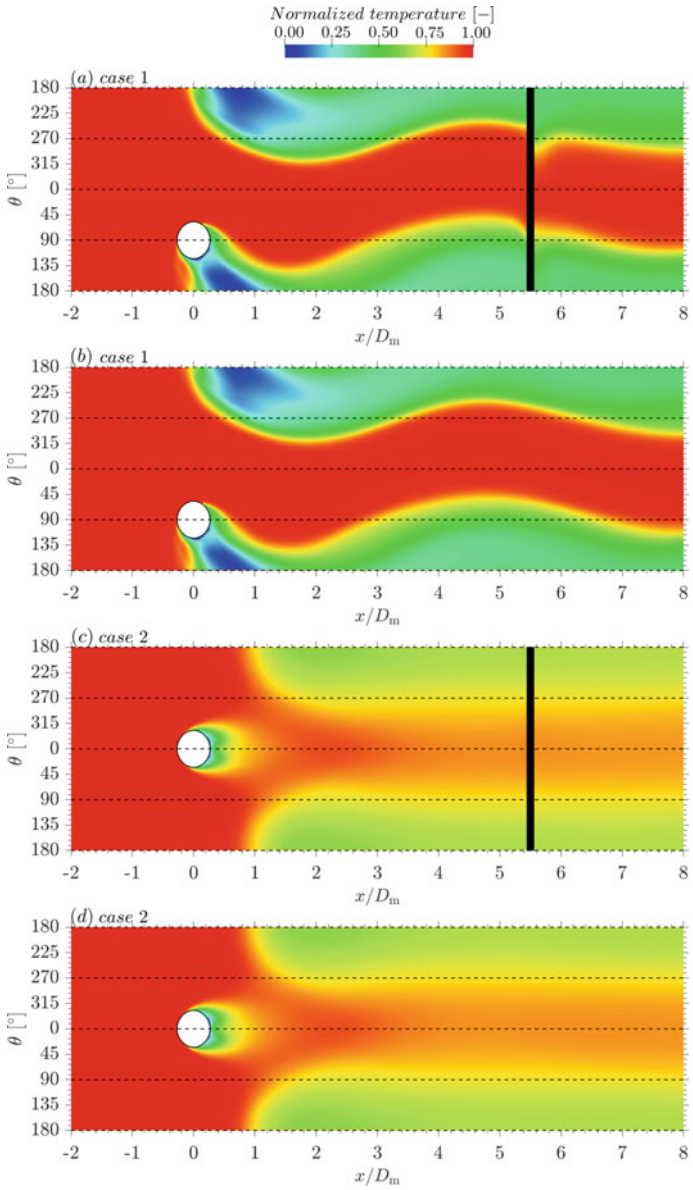


Fig. 3 Circumferential distribution of the mean normalized temperature $\overline{T^*}$ along the downstream (1 mm from the main pipe wall) with **a, c** and without **b, d** weld seam for both cases

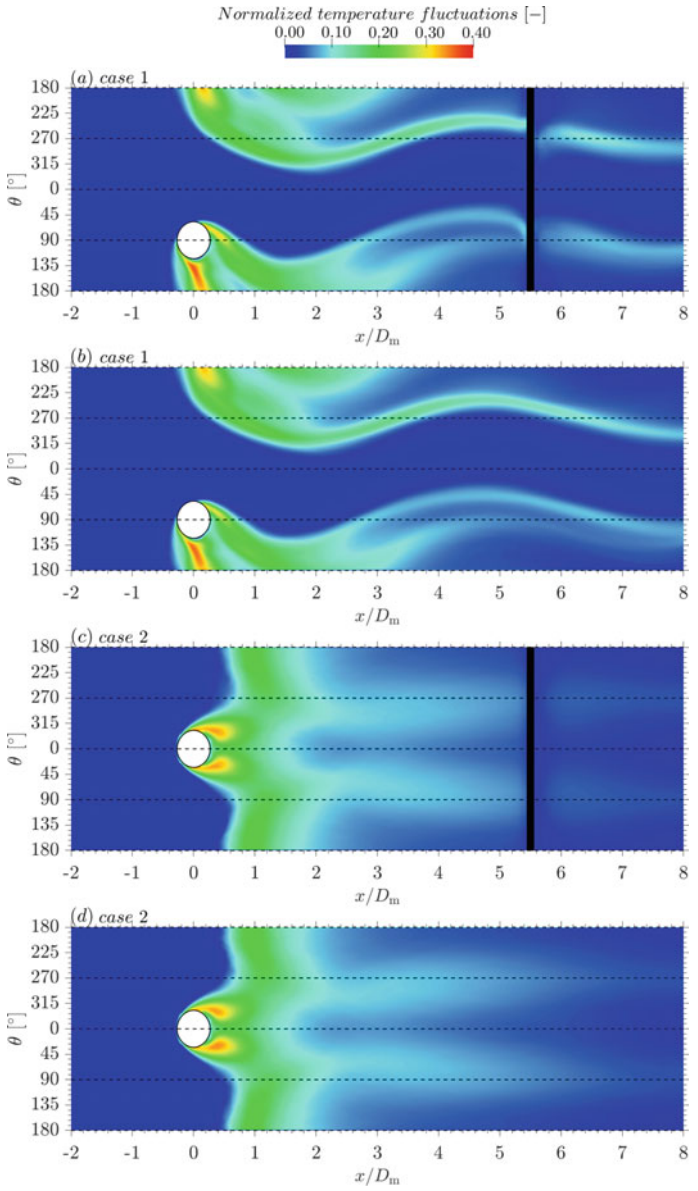


Fig. 4 Circumferential distribution of the mean normalized temperature T^*_{RMS} along the downstream (1 mm from the main pipe wall) with **a, c** and without **b, d** weld seam for both cases

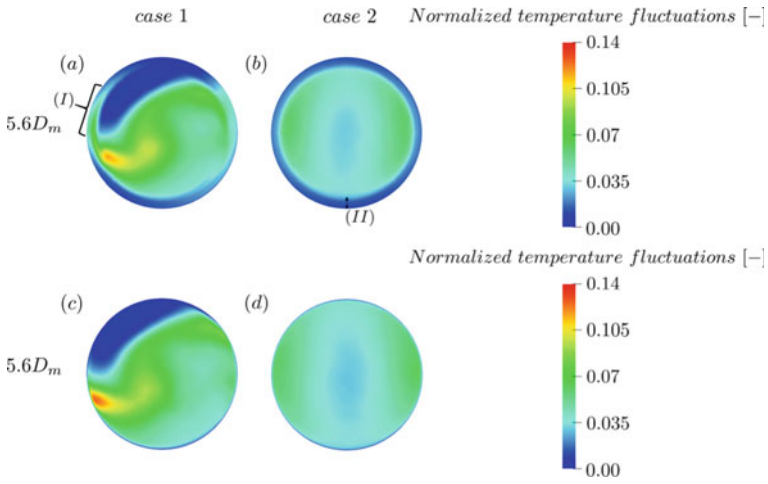


Fig. 5 Normalized temperature fluctuations at $x = 5.6D_m$ with **a, b** and without **c, d** weld seam for both cases

3.3 Flow Characteristics Close to the Weld Seam

Figure 5(a–d) show the normalized temperature fluctuations with/without weld seam at the position $x = 5.6D_m$ downstream of the T-junction for both cases.

The weld seam at the position $x = 5.5D_m$ downstream of the T-junction significantly affects the mean thermal field. The effects of the reduction of the inner diameter on the mixing behavior in case 1 can be seen in Fig. 5(a, c), where the fluctuation intensity in the near-wall region is rotated and shifted upwards in area (I). In case 2, the thermal fluctuations in the near-wall region (II) are strongly decreased as shown in Fig. 5(b, d).

3.4 Computational Performance

The scalability test of the T-junction mixing process on the high-performance computer named 'Hazel Hen', which is located at the High-Performance Computing Center (HLRS) Stuttgart, is shown in Fig. 6. The scaling performance is analyzed against the number of cores and 24 cores per node were used. The data acquisition took place at a fixed time step size. To improve the simulation time, the output of results was suppressed during each scaling test run, providing a constant condition for the investigation. The numerical meshes are subdivided into parts and the load is uniformly distributed over the cores. The hierarchical geometric method is used to decompose the domain. Ideally, the method should scale linearly as shown in Fig. 6(a). The speedup is defined as $T_{core=24}/T_{core=N}$, whereby $T_{core=24}$ represents

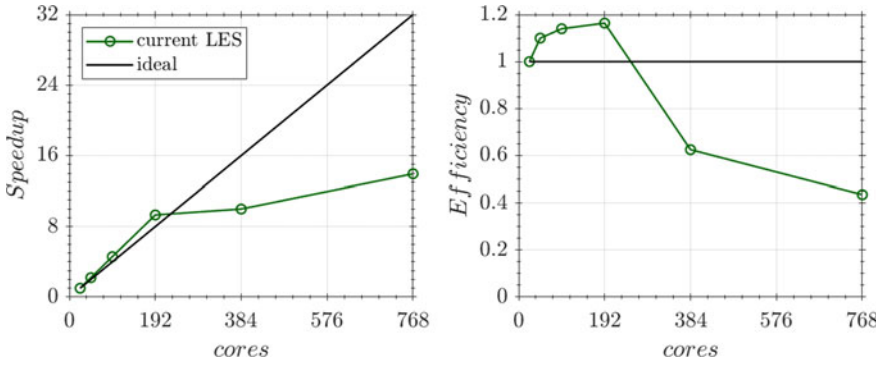


Fig. 6 Computational performance of current solver **a** speedup **b** parallel efficiency

the simulation time with 24 cores while $T_{core=N}$ defines the time using N cores with $24 \leq N \leq 768$. Similarly, parallel efficiency is defined as the ratio of wallclock speedup to the number of nodes used.

For the scalability test 100 time-steps were used to characterize the code. It took an inordinate amount of time to calculate the case on 1 core, it was approximately $t \approx 3$ h. We infer that the method has an approximately linear speedup for the investigated case until 192 cores, see Fig. 6(a). The simulation with 192 cores is around 8 times faster than with 24 cores. The simulation with 192 cores is equivalent to around 90000 cells per core. Furthermore, the simulation with 384 cores is around 10 times faster than with 24 cores and this corresponds to approximately 46000 cells per core. It can be seen that by increasing the cores to 768, the speedup drops and the parallel efficiency falls to 44%, see Fig. 6(b). The T-junction mixing processes are simulated for 180 s and this is sufficient for the demonstration case presented here. The computational effort for this kind of simulations are approximately $5 * 10^4$ CPU hours. In order to avoid the limitation of memory storage and the number of output files, we recorded only the three latest time step data in this investigation.

4 Conclusions

In this work, a numerical analysis was performed for a better understanding of stratified mixing flows in T-junction configurations. To achieve this goal, wall-resolved Large-Eddy Simulation was performed. The investigated cases have a temperature difference of 180 K between the main and branch pipe fluids, in this connection the mass flow rate ratio is 3. The Reynolds number in the main pipe is around 78400 and in the branch pipe approximately 6600. For different cases were analyzed to understand the effect of the weld seam on the mixing process and thermal fluctuations. The numerical results show a good agreement with the corresponding experimental data. Furthermore, a wavy stable stratification can be detected in the case 1. Here the

upperpart of the pipe stays warm during the mixing process. An enhanced thermal mixing can be seen in case 2. This case shows a stronger effect of buoyancy, however without an oscillating behavior or wavy flow pattern type. The thermal fluctuations field represents that the highest intensities are on the interface layer between warm and cold. Additionally, in case 1 are the thermal fluctuation intensities in the further downstream higher than in case 2. We also conclude from the simulations that the stratification layer is rotated and shifted due to the presence of the weld seam.

Acknowledgements The presented work is funded by the German Federal Ministry of Economic Affairs and Energy (BMWi, project no. 1501575) on the basis of a decision by the German Bundestag. All authors extend their sincere thanks to the High Performance Computing Center (HLRS) of the University of Stuttgart and to the Steinbuch Centre for Computing (SCC) of the University of Karlsruhe for granting access to high-performance computers for LES.

References

1. S. Chapuliot, C. Gourdin, T. Payen, J.P. Magnaud, A. Monavon, Hydro-thermal-mechanical analysis of thermal fatigue in a mixing tee. Nucl. Eng. Design **235**, 575–596 (2005). https://doi.org/10.1016/j.nucengdes.2004.09.011
2. X. Chu, E. Laurien, D.M. McEligot, Direct numerical simulation of strongly heated air flow in a vertical pipe. Int. J. Heat Mass Transf. **101**, 1163–1176 (2016). https://doi.org/10.1016/j.ijheatmasstransfer.2016.05.038
3. C. Evrim, E. Laurien, Large-Eddy simulation of turbulent thermal flow mixing in a vertical T-Junction configuration. Int. J. Therm. Sci. **150**, 106,231 (2020). https://doi.org/10.1016/j.ijthermalsci.2019.106231
4. N. Fukushima, K. Fukagata, N. Kasagi, H. Noguchi, K. Tanimoto, Numerical and experimental study on turbulent thermal mixing in a t-junction flow, in *The 6th ASME-JSME Thermal Engineering Joint Conference* (2003)
5. T. Höhne, Scale resolved simulations of the OECD/NEA-Vattenfall T-junction benchmark. Nucl. Eng. Design **269**, 149–154 (2014). https://doi.org/10.1016/j.nucengdes.2013.08.021
6. S.T. Jayaraju, E.M.J. Komen, E. Baglietto, Suitability of wall-functions in Large Eddy Simulation for thermal fatigue in a T-junction. Nucl. Eng. Design **240**, 2544–2554 (2010). https://doi.org/10.1016/j.nucengdes.2010.05.026
7. M. Kuschewski, R. Kulenovic, E. Laurien, Experimental setup for the investigation of fluid-structure interactions in a T-junction. Nucl. Eng. Design **264**, 223–230 (2013). https://doi.org/10.1016/j.nucengdes.2013.02.024
8. T.S. Lund, X. Wu, K.D. Squires, Generation of turbulent inflow data for spatially-developing boundary layer simulations. J. Comput. Phys. **140**(2), 233–258 (1998)
9. F. Nicoud, F. Ducros, Subgrid-scale stress modelling based on the square of the velocity gradient tensor. Flow Turbul. Combust. **62**, 183–200 (1999)
10. U. Piomelli, J.R. Chasnov, Large-Eddy simulations: theory and applications, in *Turbulence and Transition Modelling: Lecture Notes from the ERCOFTAC/IUTAM Summerschool held in Stockholm*. ed. by M. Hallbäck, D.S. Henningson, A.V. Johansson, P.H. Alfredsson (Springer, Dordrecht, 1996), pp. 269–336. https://doi.org/10.1007/978-94-015-8666-5_7
11. W. Schoppa, F. Hussain, Coherent structure dynamics in near-wall turbulence. Fluid Dyn. Res. **26**(2), 119–139 (2000)
12. K. Selvam, R. Kulenovic, E. Laurien, J. Kickhofel, H.M. Prasser, Thermal mixing of flows in horizontal T-junctions with low branch velocities. Nucl. Eng. Design **322**, 32–54 (2017). https://doi.org/10.1016/j.nucengdes.2017.06.041

13. B.L. Smith, J.H. Mahaffy, K. Angele, J. Westin, Report of the OECD/NEA-vattenfall t-junction benchmark exercise (2011)
14. M. Zhou, R. Kulenovic, E. Laurien, Experimental investigation on the thermal mixing characteristics at a 90 °C T-Junction with varied temperature differences. Appl. Therm. Eng. **128**, 1359–1371 (2018). https://doi.org/10.1016/j.applthermaleng.2017.09.118

Turbulence Modulation and Energy Transfer in Turbulent Channel Flow Coupled with One-Side Porous Media

Xu Chu, Wenkang Wang, Johannes Müller, Hendrik Von Schöning, Yanchao Liu, and Bernhard Weigand

Abstract The microscopic structure of porous walls modulates the turbulent flow above. The standard approach, the volume-averaged modelling of the porous wall, does not resolve the pore structure. To systematically link geometric characteristics with flow properites, direct numerical simulations are conducted which are fully-resolving the microscopic structure. A high-order spectral/*hp* element solver is adopted to solve the incompressible Navier-Stokes equations. Resolving the full energy-spectra relies on a zonal polynomial refinement based on a conforming mesh. A low and a high porosity case with in-line arrays of cylinders are analysed for two Reynolds numbers. The peak in the streamwise energy spectra is shifted towards the pore unit length for both cases. Proper Orthogonal Decomposition (POD) shows that the fluctuations in the porous wall are linked to the structures above. Q2 structures are linked with blowing events and Q4 structures with suction events in the first pore row. The numerical solver Nektar exhibits an excellent scalability up to 96k cores on "Hazel Hen" where a slightly improved performance is observed on the brand new HPE "Hawk" system. Strong scaling tests indicate an efficiency of 70% with around 5, 000 mesh-nodes per core, which indicates a high potential for an adequate use of a HPC platform to investigate turbulent flows above porous walls while resolving the pore structure.

1 Introduction

The properties of turbulent flows bounded by porous materials are determined by the microscopic structure of the porous material. This modification of the turbulent flow by a porous structure can be found in natural systems and in various engineering applications. Natural systems are for instance rivers flowing over porous riverbeds or the flow of wind over forests. In engineering the ability of a porous wall to exchange momentum and energy with the turbulent flow is useful for Aerodynamics and Ther-

X. Chu (✉) · W. Wang · J. Müller · H. Von Schöning · Y. Liu · B. Weigand
Institute of Aerospace Thermodynamics, University of Stuttgart, Pfaffenwaldring 31, 70569 Stuttgart, Germany
e-mail: xu.chu@itlr.uni-stuttgart.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_24

modynamics applications. Thermodynamics applications like heat exchangers use the high specific surface area of porous walls to increase the heat transfer. The process of transpiration cooling involves porous walls through which gas is blown to reduce the heat flux into the structure and at the same time to absorb heat from the structure. Aerodynamics investigations show, that it is possible to reduce the noise of an airfoil by applying a porous structure at the trailing edge. All these examples have in common that macroscale properties are determined by the microscopic characteristics of the porous structure or the surface structure. Reducing frictional pressure losses with an increase in heat transfer by the design of a passive structure would be of great commercial value. To enable this it is decisive to understand the influence of the characteristics of porous structures like the pore mophology and topology.

The modulation of turbulence by a permeable surface has been confirmed in early-years experiments with different configurations, e.g. turbulent open channel flows over porous media composed of spheres. A qualitative similar conclusion was drawn that the wall permeability is able to increase turbulent friction. In a recent study, Suga et al. [21] constructed three different kinds of anisotropic porous media to form the permeable bottom wall of the channel. The wall permeability tensor is designed to own a larger wall-normal diagonal component (wall-normal permeability) than the other components and the spanwise turbulent structures are investigated with particle image velocimetry (PIV). They discuss the correlation of streak spacing and integral length with the wall normal distance for different wall permeabilities.

Terzis et al. [22] examined experimentally the hydrodynamic interaction between a regular porous medium and an adjacent free-flow channel at low Reynolds numbers ($Re < 1$). In their study the porous medium consists of evenly spaced microstructured rectangular pillars arranged in a uniform pattern, while the free-flow channel features a rectangular cross-sectional area. They show that the flow is non-parallel at the interface between the free-flow and the porous media flow. DNS allows microscopic visualization and analysis, which is hardly achvieable within the measurements in such confined and tortuous spaces. The existing experiments provide information about the optically accessible areas. However, because of the difficulty in performing measurements inside the porous media, it is not easy to discuss the turbulent flow physics inside the porous structures.

Direct Numerical Simulation exhibits an edge of observing and analyzing turbulent physics in a confined small space, not only for the canonical cases as channel flows and pipe flows [2, 4, 13–15, 17, 19], but also for transitional and turbulent flow in a representative elementary volume (REV) of porous media [5, 7]. Growing interest has been observed regarding the turbulent flow regime inside porous structures, as the studies with direct numerical simulation (DNS) show [5–7, 11, 26]. However, an adequate resolution of the smallest length scales of the flow in the interface region (between the pore flow and the turbulent boundary layer) requires enormous computational resources. To limit the computational cost, the size of the computational domain can be reduced. Therefore, appropriate boundary conditions must be chosen. Jimenez et al. [10] performed DNS with a special boundary condition: They imposed no-slip conditions for the streamwise and spanwise velocities, and set the wall-normal velocity for the permeable wall to be proportional to the local

pressure fluctuations. The friction is increased by up to 40% over the walls, which was associated with the presence of large spanwise rollers.

Rosti et al. [20] explored the potential of drag reduction with porous materials. They systematically adjusted the permeability tensor on the walls of a turbulent channel flow via VANS-DNS coupling. The total drag could be either reduced or increased by more than 20% through adjusting the permeability directional properties. Configuring the permeability in the vertical direction lower than the one in the wall-parallel planes led to significant streaky turbulent structures (quasi 1-dimensional turbulence) and hence achieved a drag reduction. Recent studies achieved to resolve the porous media structures coupled with turbulent flows. Kuwata et al. [12] used the Lattice-Boltzmann method (LBM) to resolve porous structures coupled with turbulent flows. The porous media is composed by interconnected staggered cube arrays. The difference between a rough wall and a permeable wall is elucidated.

The current study is intended to establish interface-resolved DNS research about turbulent flows over porous media. Through an adequate resolution of the flow field inside the porous structure and of the turbulent flow above, both the turbulence modulation and the energy exchange across the porous surface is investigated. This physical knowledge can be used to support different levels of modeling like LES or RANS [28]. Furthermore, it will be possible to link the geometrical characteristics of the porous media with the turbulent structures of the flow field. This will enable the design of porous structures that generate specific flow properties.

2 Numerical Method

2.1 Arbitrary High-Order Numerical Solver for Complex Geometries

The three-dimensional incompressible Navier−Stokes equations, given by Eqs. 1–2, are solved in non-dimensional form, where Π is the corresponding source term in the momentum equation to maintain a constant pressure gradient in the main flow direction.

$$\frac{\partial u_j}{\partial x_j} = 0 \tag{1}$$

$$\frac{\partial u_i}{\partial t} + \frac{\partial u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{1}{Re}\frac{\partial^2 u_i}{\partial x_i \partial x_j} + \Pi \delta_{i1} \tag{2}$$

A spectral/*hp* element solver Nektar++ [1, 18] is used to perform the DNS and to resolve the wide-spectra of scales in the complex geometrical structures. The high-order solver framework allows arbitrary-order spectral/*hp* element discretisations and refinement with hybrid shaped elements. Both modal and nodal polynomial functions can be used for the high-order representation. In addition, the homogeneous flow direction can be represented with Fourier-spectral expansions and, therefore,

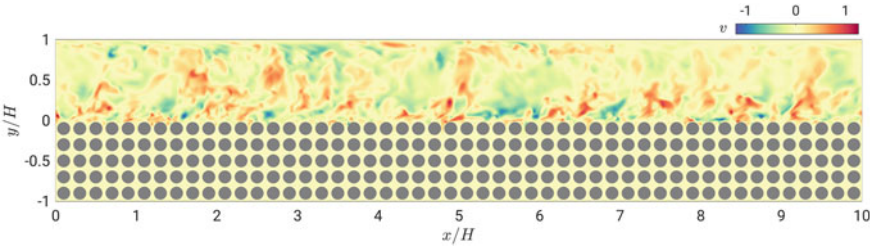


Fig. 1 Interface resolved Direct Numerical Simulation. Section of the computational domain in the $x - y$ plane. The spanwise direction is periodic. Depicted is a snapshot of the streamwise velocity fluctuation u'.

enables the usage of an efficient parallel direct solver. The spectral-accurate discretisation combined with meshing flexibility is optimal to deal with complex porous structures and to resolve the interface region. The time-stepping is treated with a second-order mixed implicit-explicit (IMEX) scheme. The fixed time step is defined by $\Delta T/(h/\overline{u}) = 0.0001$ to ensure numerical stability. A run of 10 flow through times for flow development and 5 flow through times for gathering statistics is in general necessary.

Figure 1 illustrates a two-dimensional sketch of the simulation domain, the spanwise direction is periodic. The domain size $L_x \times L_y \times L_z$ is $100 \times 20 \times 8\pi$, where the lower part $0 < y < 10$ is the porous media and the upper part $10 < y < 20$ is the free-flow channel. The porous layer consists of circular cylinders arranged in-line. The porous layer consists of 50 porous media elements in streamwise direction and 5 elements in wall-normal direction, which indicates 250 porous elements in total. A no-slip boundary condition is defined for all surfaces of the porous structure (i.g. on the surfaces of the circular cylinders) as well as for the upper wall ($y = 20$) and for the lower wall ($y = 0$). Periodic boundary conditions are defined in x-direction between $x = 0$ and $x = 100$. The second pair of periodicity is defined in the spanwise direction $z = 0$ and $z = 8\pi$.

The geometry is discretizied with full quadrilateral elements on the $x - y$ plane with local refinement near the permeable interface. The third direction (z-direction) is extended with a Fourier-based spectral method. High-order Lagrange polynomials through the modified base are applied on the $x - y$ plane. The numerical solver enables a flexible non-identical polynomial order based on the conforming elements, which offers high meshing flexibility corresponding to laminar, turbulent or interfacial flows according to prior knowledge.

2.2 Simulation Conditions

The simulation cases are summarized in Table 1. Four cases are introduced here covering two porous topologies and two Reynolds numbers. The porous topology

Table 1 Summary of boundary layer parameters for all the cases. Case A1 and A2 are the low porosity cases ($\varphi = 0.5$) and case B1 and B2 are the high porosity cases ($\varphi = 0.8$)

Case	φ	U_b	ν	Re	u_τ^p	Re_τ^p	u_τ^t	Re_τ^t
A1	0.5	5.15	5.92e−3	3680	0.32	335	0.28	193
B1	0.8	3.58	5.92e−3	2519	0.41	503	0.28	123
A2	0.5	5.45	2.96e−3	7804	0.31	667	0.28	367
B2	0.8	3.58	2.96e−3	5088	0.41	979	0.28	231

is characterised by the porosity φ, which is defined by the ratio of the void volume V_V to the total volume V_T of the porous structure. The bulk Reynolds number is defined using the bulk averaged velocity $\overline{u}$, channel height h and kinematic viscosity ν, i.e. $Re = \overline{u}h/\nu$. Additional flow characteristics are the shear Reynolds numbers $Re_\tau^t = u_\tau^t h/\nu$ and $Re_\tau^p = u_\tau^p h/\nu$. Wherein the superscript t refers to the evaluation at the top wall of the channel and p to the evaluation at the permeable surface. The total mesh resolution ranges from 250×10^6 to 1.1×10^9 for the high Reynolds number condition.

3 Results

Figure 2 shows the mean streamwise velocity U profiles, where $U = \langle \overline{u} \rangle$ is temporally and spatially averaged. The inner-scaled U^+ profiles of both sides are compared in Fig. 2(a), which are normalized by u_τ^p and u_τ^t correspondingly. All the velocity profiles on the smooth top wall (dashed lines) follow the linear and log law fairly well regardless their difference in porosities and Reynolds numbers, which indicates a marginal influence of the porous media on this side. On the other hand, the velocity profiles above the permeable wall differ significantly from the canonical boundary layer profiles. The U^+ profiles of the porous media side are much lower than the smooth wall side, owing to the increase of the friction velocity u_τ^p. Moreover, the friction velocity u_τ^p shows an increase around 32% from the low porosity cases (A1 and A2) to the high porosity ones (B1 and B2), resulting in a much lower magnitude of U^+ for the latter.

The velocity profiles U^{t+} in the entire domain are shown in Fig. 2(b), which are normalized by the friction velocity of the smooth wall side u_τ^t. There are several important differences between the low and the high porosity cases. First, the thickness of the boundary layer is larger for walls with higher permeability, indicating a larger impact area in the channel. Second, the velocity within the porous media increases with porosity φ. Third, the velocity in the vicinity of the interface is negative with a small value for cases A1 and A2, which is related to the recirculation region between the cylinders. In contrast, the velocity above the smaller cylinders (cases B1 and B2) is monotonic, indicating an absence of a rotational region. This will be more

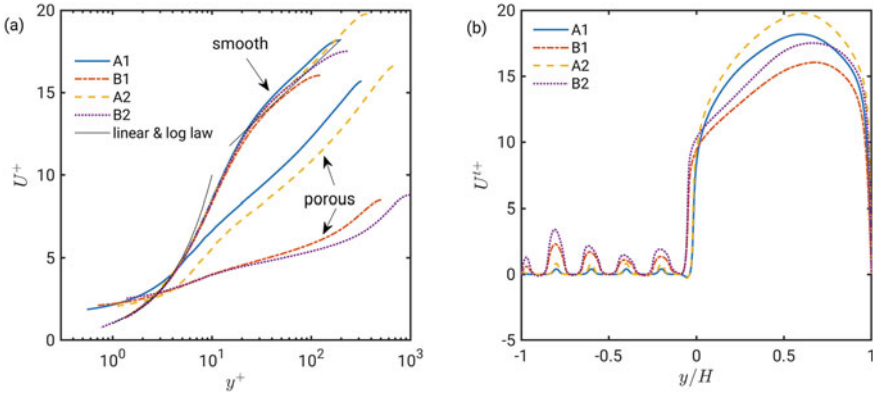


Fig. 2 Mean streamwise velocity profiles U. **a** U^+ profiles above the porous wall and near the smooth wall normalized by their own inner variables; **b** U^{t+} profiles of the whole domain normalized by u_τ^t

clearly shown in the later discussion. Increasing the Reynolds number, the magnitude of the U^{t+} within the porous media is enhanced, which may indicate more active momentum exchange near the interface.

The contours of the time averaged velocities $\overline{u}$ and $\overline{v}$ of the interface region are depicted in Fig. 3. A pair of counter-rotating vortices are observed between two circular cylinders in the low-porosity cases A1 and A2 Fig. 3(a, c), which is absent in the high-porosity cases Fig. 3(b, d). The recirculation induced by the vortices leads to the backflow near the interface of the U profile Fig. 2(b). The upper vortex is driven by the main stream velocity and restricted by the narrow void between cylinders, which results in a classical lid-driven cavity flow here. The upper and lower vortex are separated by the narrow throat between the cylinders, which blocks the convection from below. In comparison, the mean streamwise and vertical velocity within the porous media are higher in case B1 Fig. 3(b) and B2 Fig. 3(d). Between two neighboring cylinders, a blow event (positive $\overline{v}$) is followed by a suction event (negative $\overline{v}$) in the downstream direction, which exchanges fluid between the interface and positions below the cylinder.

The wall normal variation of turbulent kinetic energy (TKE) and Reynolds shear stress, i.e., $\langle \overline{u_i' u_j'} \rangle_s^{t+}$, are depicted in Fig. 4, which are averaged in x-z plane and normalized by u_τ^t. Note that the subscript s denotes superficial area averaging. For the cases studied, all the components of the TKE and the Reynolds shear stress are intensified on the porous media side compared to its counter part of the non-permeable wall, and the high porosity cases (B1, B2) have even a higher magnitude than the lower ones. Increasing the Reynolds number results in a further increment of all four Reynolds stress components.

Despite sharing similar features above, the high and low porosity cases show clear differences at the interface and below. For cases A1 and A2, the TKE components approach zero quickly as moving from the permeable interface into the porous media

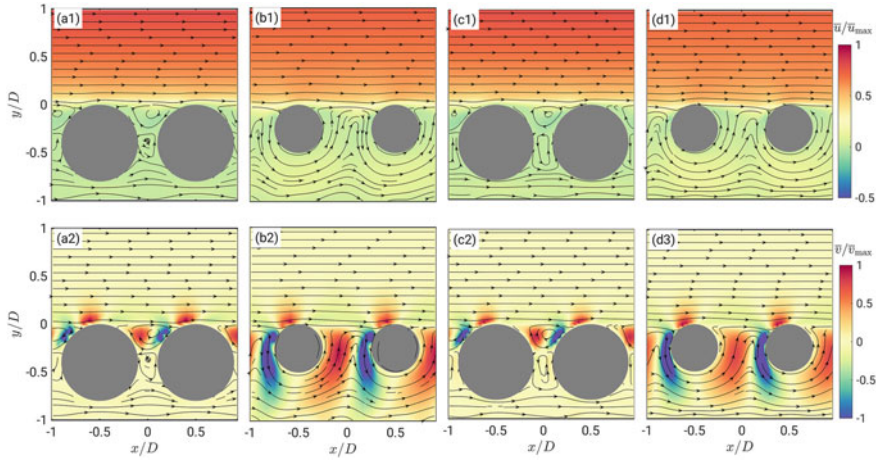


Fig. 3 Ensemble averaged velocity fields $\overline{u}$ (row 1) and $\overline{v}$ (row 2). Column a: case A1; Column b: case B1; Column c: case A2; Column d: case B2. The wall normal origin is set at the crest of the cylinders

domain, which suggests that disturbances in the free-flow result only in a marginal penetration into the porous media domain. In contrast, turbulent fluctuations are still relatively energetic below the interface for cases B1 and B2. The streamwise component $\langle\overline{u'u'}\rangle_s^{t+}$ shows a periodic distribution in the porous media domain due to the blockage of the cylinders, while the other two components show a smooth descending trend as moving towards the bottom wall. Nevertheless, the Reynolds shear stress $\langle\overline{u'v'}\rangle_s^{t+}$ becomes weak below the first row of cylinders for all the cases.

To show the spatial variation of TKE and Reynolds shear stress, contours of $\overline{u'u'}$, $\overline{v'v'}$, $\overline{w'w'}$, $\overline{u'v'}$ close to the permeable interface are depicted in Fig. 4. The magnitude of $\overline{u'u'}$ fades quickly below the porous bed in all four cases. As for $\overline{v'v'}$, $\overline{w'w'}$, and $\overline{u'v'}$, the momentum flux represented by these terms have a deeper impact region in the high porosity cases. A positive peak of the Reynolds shear stress $\overline{u'v'}$ is observed at the impinging position B in all four cases, which differs from the negative Reynolds shear stress above the porous bed. For cases B1 and B2, an area of positive Reynolds shear stress reaches even below the first layer of cylinders.

In addition to the one-point statistics, the porous media may also change the energetic scale in the free-flow, which further influences the sustaining process of turbulence. Figure 5 shows one-dimensional pre-multiplied spectra of turbulent kinetic energy $k_x\hat{q}/k_z\hat{q}$ as a function of the streamwise/spanwise wave lengths λ_x, λ_z and wall distance y^{t+}. Here, $\hat{\cdot}$ stands for the Fourier coefficients that have been transformed in x- or z-direction. The wave length λ_x and λ_z are normalized with the distance between two cylinders (pore unit length) D. The streamwise spectra is averaged in time and spanwise dimension, while the spanwise spectra is averaged in time and streamwise dimension. The spectra above the non-permeable wall are flipped

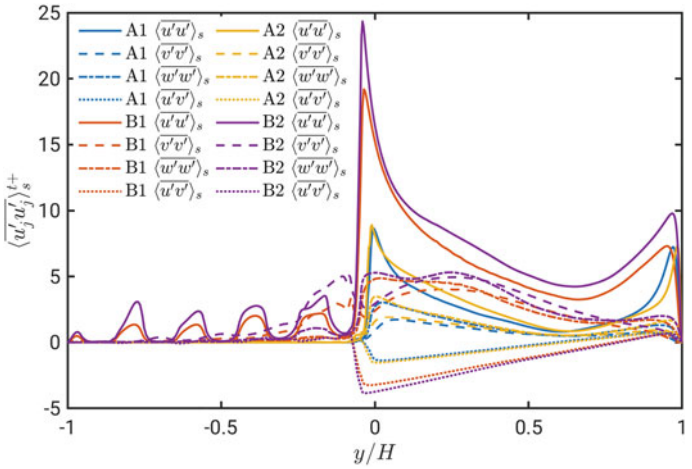


Fig. 4 Superficial area averaged Reynolds stresses $\langle\overline{u_i'u_i'}\rangle_s^{t+}$ normalized by the friction velocity u_τ^t

upside down and superimposed (as solid lines) on the spectra above of permeable wall (color contour).

A series of high-energy spikes are observed in the streamwise spectra of all cases, which are originated from the porous wall and reach as far as $y^{t+} \approx 60$. The wave length of the spikes feature a series of harmonic waves with a maximal wave length of the pore unit length D, which indicates that these spikes stand for the highly regulated fluctuation stemming from the porous medium. In addition to the spikes, the remaining of the spectra represent the energy of 'background' turbulence. In cases A1 and A2, the peak of the 'background' spectra above the porous wall is $\lambda_x^{t+} \approx 150$ in inner scale at $y^{t+} \approx 20$, which is slightly smaller than $\lambda_x^{t+} \approx 200$ of the smooth wall side. In case A2, with higher Reynolds number, there is a strong trend that the peak the 'background turbulence' spectra is synchronized with the porous unit spikes. The energy concentrates between $\lambda_x \approx 2D$ or $\lambda_x^{t+} \approx 400$ and $\lambda_x \approx 0.2D$ or $\lambda_x^{t+} \approx 40$. For the cases B1 and B2, the turbulent spectra also bias towards spikes at $\lambda_x \approx 1 \sim 2D$ obviously, especially for the high Reynolds number case B2. Moreover, energetic peaks at large scales $\lambda_x \geq 10D$ are observed for spectra on both sides. It appears that the periodic fluctuations originated from the permeable wall perform as an additional source in the energy cascade of turbulence. By introducing additional energetic modes into the spectra, the porous wall can efficiently biased the peak of streamwise spectra towards the pore unit length.

The TKE spectra show a significant impact of the porous medium on scale energy, especially for the streamwise modes. This indicate that the coherent structures above the porous media can be quite different from those of a canonical wall-bounded flow in terms of length scale and evolution dynamics. Further effort is needed to shed light on this problem.

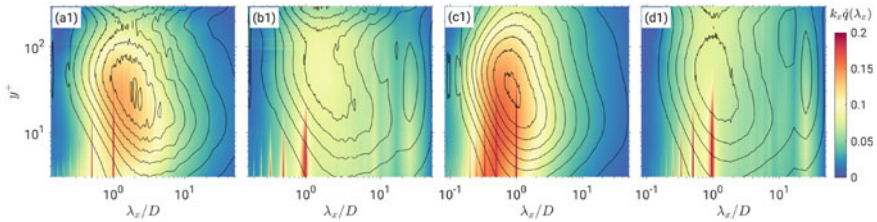


Fig. 5 Pre-multiplied one-dimensional TKE spectra: $k_x\widehat{q}(\lambda_x)$. Column a–b represent the cases A1, B1, A2, and B2, respectively. The color contours are the energy spectra from the porous media side, while the solid isolines representing spectra of the non-permeable side which are superimposed as a reference

Proper orthogonal decomposition (POD, [23–25]) is applied to the flow fields to extract the most energetic structures. The region is selected to be from the first layer cylinders to the middle of the channel. The first six spatial modes are shown in Fig. 6 The turbulent fluctuation in porous media is one order smaller than that above the interface. The POD modes mainly capture the coherent structure in the TBL. The first two modes are coupled, with large scale Q2 and Q4 structures connected in streamwise direction. These Q2/Q4 zones have a streamwise extent of $\Delta X \geq 20$, and reach beyond the center of the channel in wall normal direction. Note that there are also smaller scale Q2/Q4 events close to the interface. Modes 3–6 represent coherent structures of smaller scale, which is similar with the results of canonical TBLs. The fluctuation with in the porous media, despite of its small contribution to TKE, has a strong connection with the structures above the interface. It is clearly shown in modes 1 and 2 that large scale Q4 structures are associated with negative v' (i.e., suction) events at the gap between the cylinders, while Q2 events with positive v' (i.e., blow) events. For high order modes, this association is not observed. This provides first-hand information about the interaction mechanism between free-flow and fluid inside the porous medium, which suggests the blow and suction at the interface is more subject to the influence of large scale motions. Figure 7 shows the cumulative energy curve. The first two modes only account for 20% of the total TKE, which is within our exception, since the turbulence above the interface is quite close to a channel flow above a smooth wall. Further POD analysis will be conducted close to the interface region to reveal more about the small scale interaction dynamics.

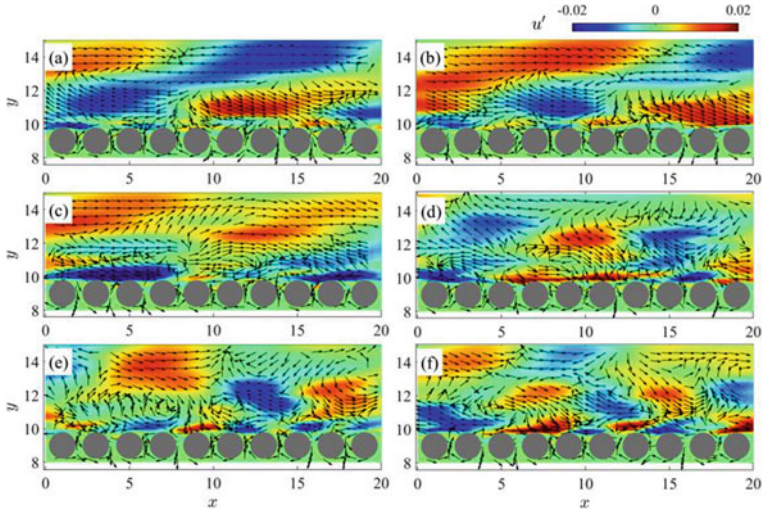


Fig. 6 The first six POD modes from DNS snapshots of coupled porous media and channel flow

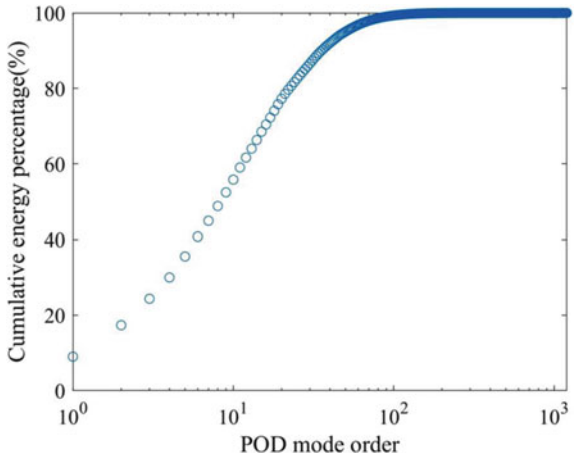


Fig. 7 Accumulated modal energy from the first 1000 POD modes

4 Computational Performance

In this section, the parallel computational performance will be discussed. The supercomputing systems utilized were 'Hazel Hen' and 'HAWK' located at the High-Performance Computer Center Stuttgart (HLRS). 'Hazel Hen', a Cray XC40 system, consists of 7,712 compute nodes. Two Intel Haswell processors (E5-2680 v3, 12 cores) and 128 GB memory are deployed at each node. The compute nodes are interconnected by a Cray Aries network with a Dragonfly topology. This amounts to a total of 185,088 cores and a theoretical peak performance of 7.4 petaFLOPS.

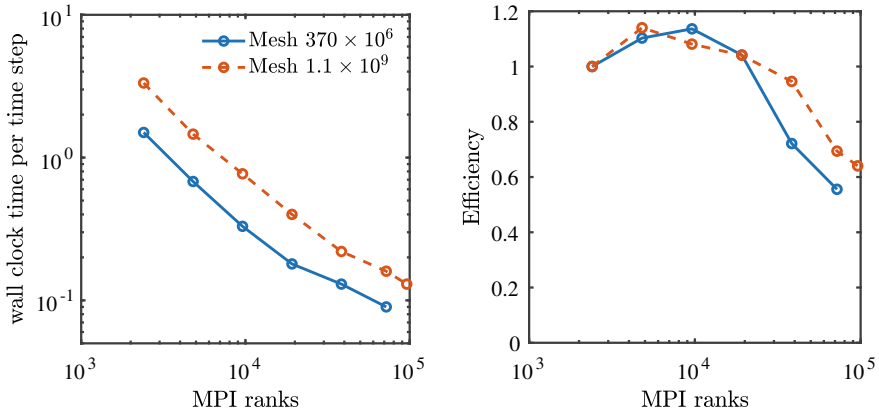


Fig. 8 Scaling behavior of the solver on 'Hazel Hen', wall clock time per time step on the left side. On the right side the efficiency for different numbers of MPI ranks

The new flagship machine HAWK, based on HPE platform running AMD EPYC 7742 processor code named Rome, will have a theoretical peak performance of 26 petaFLOPs, and consist of a 5,632-node cluster. The AMD EPYC 7742 CPU consists of 64 cores, which leads to 128 cores on a single node sharing 256 GB memory on board. This means that a total of 720,896 cores are available on the HAWK system.

The scalability tests based on two cases are shown in Fig. 8, one with 370×10^6 mesh element nodes and the other one with 1.1×10^9. Each MPI rank in the simulations represents one core on Hazel Hen. The total number of MPI ranks tested on the machine ranges from 2,400 to 96,000. A lower number of ranks is not possible due to the memory overload. An approximate linear relationship between the number of MPI ranks and the wall clock time per timestep can be identified on the left hand side. One time step takes about 0.08 s with a load of around 5,000 DOF per core. The strong scaling efficiency on the right hand side exhibits a super linear speedup at low MPI ranks, followed by a slight drop for more than 10^4 MPI ranks. A minimum efficiency of 70% can be achieved with the maximal number of MPI ranks. Compared to the previously employed open-source FVM code OpenFOAM [3, 4, 8, 16, 27] and the discontinuous-Galerkin spectral-element method (DGSEM) code [9], the open-source spectral/hp element code shows strong scalability on the HPC machine. Due to the recent status of HAWK, the planned scalability test was not conducted. According to real-field usage experience, the per-core performance on HAWK is slightly better than that on Hazel Hen (approximately 20%). The scalability is expected to be at least on the same level of Hazel Hen.

5 Conclusions

We studied turbulent flow over porous media by conducting fully-resolved direct numerical simulations of two porous topologies and two Reynolds numbers cases. The porous wall leads to an asymmetric velocity profile in the channel. This asymmety is more pronounced for higher porosites and leads to a larger boundary layer thickness and lower velocities on the porous wall side. The friction velocity is increased around 32% for a higher porosity ($\varphi = 0.8$) compared to the lower porosity case ($\varphi = 0.5$). But also the low porosity case ($\varphi = 0.5$) has an approximatelya 10 % higher friction velocity than a smooth wall.

Furthermore, the low porosity cases have a lower velocity within the porous media but also recirculation regions between the first row of cylindes. This recirculation can be compared to classical lid-driven cavity flows which blocks the direct convection between the turbulent flow and the first row of pores.

By increasing the Reynolds number, the magnitude of the U^{t+} within the porous medium is enhanced, which indicates a higher momentum near the interface. The TKE and Reynolds shear stress demonstrate that high porosity as well as high Reynolds numbers result in a higher magnitude. For low porosity cases, the turbulent disturbances in the free-flow remains marginal below the interface, while the disturbances are still energetic in the porous media domain for high porosity cases. But these disturbances only penerate the first row of pores where they are one order smaller than in the free turbulent flow. The TKE spectra shows a significant impact of the porous structures on the scale energy, especially for the streamwise modes. For all cases, the turbulent spectra bias towards spikes at $\lambda_x \approx 1 \sim 2D$ and are stronger for high Reynolds number cases. Additional energetic peaks at large scales can be found in high porosity cases. The POD analysis shows that the fluctuation within the porous wall, which have a small contribution to the TKE, are linked to the structures above the interface. It is shown that the Q2 structures are linked to blowing events and Q4 structures are linked to suction events in the first row of pores.

We also tested the computational performance of the supercomputing system 'Hazel Hen' for the current cases. Scalability test show that the maximum efficiency is achieved at 10^4 MPI ranks and a minimum efficiency of 70% at 96,000 MPI ranks.

Acknowledgements This work is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation)—Project Number 327154368—SFB 1313. It is supported by MWK (Ministerium für Wissenschaft und Kunst) of Baden-Württemberg as a part of the project DISS (Data-integrated Simulation Science). The authors gratefully appreciate the access to the high performance computing facility 'Hazel Hen' and 'HAWK' at HLRS, Stuttgart and would like to thank the teams of HLRS, Cray and HPE for their kind support.

References

1. C.D. Cantwell, D. Moxey, A. Comerford, A. Bolis, G. Rocco, G. Mengaldo, D. De Grazia, S. Yakovlev, J.E. Lombard, D. Ekelschot et al., Nektar++: an open-source spectral/HP element framework. Comput. Phys. Commun. **192**, 205–219 (2015)
2. X. Chu, W. Chang, S. Pandey, J. Luo, B. Weigand, E. Laurien, A computationally light data-driven approach for heat transfer and hydraulic characteristics modeling of supercritical fluids: From dns to dnn. Int. J. Heat Mass Transf. **123**, 629–636 (2018)
3. X. Chu, E. Laurien, D.M. McEligot, Direct numerical simulation of strongly heated air flow in a vertical pipe. Int. J. Heat Mass Transf. **101**, 1163–1176 (2016)
4. X. Chu, E. Laurien, S. Pandey, Direct numerical simulation of heated pipe flow with strong property variation, in: *High Performance Computing in Science and Engineering 2016* (Springer, Heidelberg, 2016), pp. 473–486
5. X. Chu, B. Weigand, V. Vaikuntanathan, Flow turbulence topology in regular porous media: from macroscopic to microscopic scale with direct numerical simulation. Phys. Fluids **30**(6), 065,102 (2018)
6. X. Chu, Y. Wu, U. Rist, B. Weigand, Instability and transition in an elementary porous medium. Phys. Rev. Fluids **5**(4), 044,304 (2020)
7. X. Chu, G. Yang, S. Pandey, B. Weigand, Direct numerical simulation of convective heat transfer in porous media. Int. J. Heat Mass Transf. **133**, 11–20 (2019)
8. C. Evrim, X. Chu, E. Laurien, Analysis of thermal mixing characteristics in different t-junction configurations. Int. J. Heat Mass Transf. **158**, 120,019 (2020)
9. F. Föll, S. Pandey, X. Chu, C.D. Munz, E. Laurien, B. Weigand, High-fidelity direct numerical simulation of supercritical channel flow using discontinuous Galerkin spectral element method, in: *High Performance Computing in Science and Engineering'18* (Springer, Heidelberg, 2019), pp. 275–289
10. J. Jimenez, M. Uhlmann, A. Pinelli, G. Kawahara, Turbulent shear flow over active and passive porous surfaces. J. Fluid Mech. **442**, 89–117 (2001)
11. Y. Jin, M.F. Uth, A.V. Kuznetsov, H. Herwig, Numerical investigation of the possibility of macroscopic turbulence in porous media: a direct numerical simulation study. J. Fluid Mech. **766**, 76–103 (2015)
12. Y. Kuwata, K. Suga, Transport mechanism of interface turbulence over porous and rough walls. Flow Turbul. Combust. **97**(4), 1071–1093 (2016)
13. D.M. McEligot, X. Chu, J.H. Bae, E. Laurien, J.Y. Yoo, Some observations concerning "laminarization" in heated vertical tubes. Int. J. Heat Mass Transf. **163**, 120,101 (2020)
14. D.M. McEligot, X. Chu, R.S. Skifton, E. Laurien, Internal convective heat transfer to gases in the low-reynolds-number "turbulent" range. Int. J. Heat Mass Transf. **121**, 1118–1124 (2018)
15. S. Pandey, X. Chu, E. Laurien, Investigation of in-tube cooling of carbon dioxide at supercritical pressure by means of direct numerical simulation. Int. J. Heat Mass Transf. **114**, 944–957 (2017)
16. S. Pandey, X. Chu, E. Laurien, Numerical analysis of heat transfer during cooling of supercritical fluid by means of direct numerical simulation, in: *High Performance Computing in Science and Engineering'17* (2018), pp. 241–254
17. S. Pandey, X. Chu, E. Laurien, B. Weigand, Buoyancy induced turbulence modulation in pipe flow at supercritical pressure under cooling conditions. Phys. Fluids **30**(6), 065,105 (2018)
18. S. Pandey, X. Chu, B. Weigand, E. Laurien, J. Schumacher, Relaminarized and recovered turbulence under nonuniform body forces. Phys. Rev. Fluids **5**(10), 104,604 (2020)
19. S. Pandey, E. Laurien, X. Chu, A modified convective heat transfer model for heated pipe flow of supercritical carbon dioxide. Int. J. Therm. Sci. **117**, 227–238 (2017)
20. M. Rosti, L. Brandt, A. Pinelli, Turbulent channel flow over an anisotropic porous wall-drag increase and reduction. J. Fluid Mech. **842**, 381–394 (2018)
21. K. Suga, Y. Nakagawa, M. Kaneda, Spanwise turbulence structure over permeable walls. J. Fluid Mech. **822**, 186–201 (2017)

22. A. Terzis, I. Zarikos, K. Weishaupt, G. Yang, X. Chu, R. Helmig, B. Weigand, Microscopic velocity field measurements inside a regular porous medium adjacent to a low reynolds number channel flow. Phys. Fluids **31**(4), 042,001 (2019)
23. W. Wang, C. Pan, J. Wang, Quasi-bivariate variational mode decomposition as a tool of scale analysis in wall-bounded turbulence. Exp. Fluids **59**(1), 1 (2018)
24. W. Wang, C. Pan, J. Wang, Multi-component variational mode decomposition and its application on wall-bounded turbulence. Exp. Fluids **60**(6), 95 (2019)
25. W. Wang, C. Pan, J. Wang, Wall-normal variation of spanwise streak spacing in turbulent boundary layer with low-to-moderate reynolds number. Entropy **21**(1), 24 (2019)
26. B.D. Wood, X. He, S.V. Apte, Modeling turbulent flows in porous media. Annual Review of Fluid Mechanics **52**(1), null (2020)
27. G. Yang, X. Chu, V. Vaikuntanathan, S. Wang, J. Wu, B. Weigand, A. Terzis, Droplet mobilization at the walls of a microfluidic channel. Phys. Fluids **32**(1), 012,004 (2020)
28. G. Yang, B. Weigand, Investigation of the Klinkenberg effect in a micro/nanoporous medium by direct simulation Monte Carlo method. Phys. Rev. Fluids **3**(4), 044,201 (2018)

Simulation of Flow over Pseudo-Random Rough Surfaces

J. Yang, A. Stroh, and P. Forooghi

Abstract Direct Numerical Simulation (DNS) is used to study fully developed turbulent flow over a number of irregular rough surfaces in periodic plane channels. DNS in minimal channels, initially proposed by Chung et al. [7], is examined and generalized in the present research. To this end the DNS results for flow in channels with reduced streamwise and spanwise sizes (so-called minimal channels) are compared to those in large channels. Generation of roughness is based on a mathematical algorithm in which the height power spectrum of the roughness along with its height probability distribution can be prescribed (hence the term "pseudo-random" rough surface). This approach allows generation of roughness samples that may be considered as surrogates of realistic roughness and can replace the costly process of scanning industrial surfaces.

1 Introduction

Flows bounded by rough surfaces are abundant in a variety of engineering applications. The roughness occurs for a wide range of reasons e.g. erosion, fouling, corrosion and ice accretion. Roughness topographical properties are reported to be highly dependent on the loading history as well as the operating condition of the equipment. Resulting rough surfaces possess a great variety of roughness property. It is widely reported that the presence of roughness alters flow behaviour in the vicinity of rough walls. For instance, roughness can trigger an intensive exchange of momentum to the flow thus increase skin friction. On the other hand, due to the enhanced exchange of mass, heat exchange ability over rough walls shows significant deviation in comparison to flows over smooth walls [1]. For engineering purpose,

J. Yang · A. Stroh (✉)
Institute of Fluid Mechanics (ISTM), Karlsruhe Institute of Technoligy (KIT), Kaiserstr. 10, 76131 Karlsruhe, Germany
e-mail: Alexander.stroh@kit.edu

P. Forooghi
Department of Mecahnical and Production Engineering, Aarhus University, Inge Lehmanns Gade 10, 8000 Aarhus, Denmark

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_25

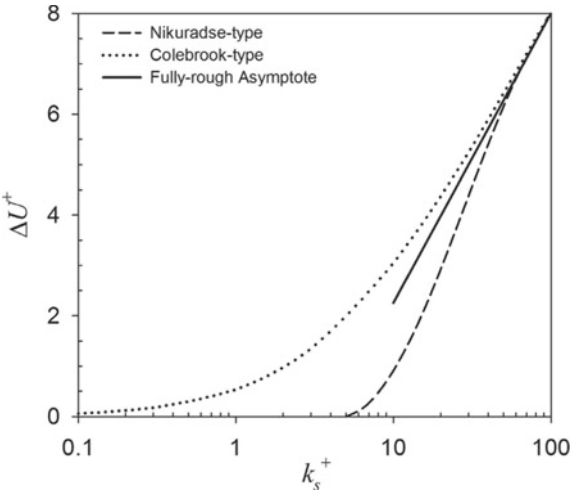


Fig. 1 Roughness function for different roughness types. Adapted from [12]

integral properties of the flow field e.g. skin friction and Nusselt number are of central interest. It is necessary to be able to predict the roughness-induced alteration in those flow properties directly from roughness geometry rather than doing experiment on every rough surface of interest, given that roughness topography differs flow properties to a considerable extent [2]. Prediction of these flow properties depending solely on the topography of roughness is since long desired. Developing the universal correlation poses its own difficulty due to the low efficiency of gathering roughness samples.

It is well established that the presence of roughness leads to an offset of the mean velocity profile. This downward shift, also called (Hama) roughness function, is recognized to be constant in logarithmic layer, which is denoted as ΔU^+ [4]. The log law over rough wall writes [3]

$$U^+ = \frac{1}{\kappa} ln(y^+) + B - \Delta U^+ \tag{1}$$

where $\kappa \approx 0.4$ is the von Karman constant, B is the log-law intercept for a smooth wall, superscript $+$ indicates wall unit scaling. ΔU^+ is a function of Reynolds number, two roughness function are shown in Fig. 1 three regimes can be observed, namely "hydraulically smooth regime, "transitionally rough" regime and "fully rough" regime. In "hydraulically smooth" regime, no roughness effect can be observed i.e. $\Delta U^+ \approx 0$. With the increase of Reynolds number "transitionally rough" regime occurs, where ΔU^+ increases with Reynolds number. As the Reynold number keep increasing, ΔU^+ reaches asymptotic line labeled as "fully rough" regime. From the work by Nikuadse [3], pipe roughened by uniform sand grain particles is studied. The correlation of sand grain size k_s and friction factor λ is then established shown in Fig. 2. For arbitrary surfaces, k_s is equivalent sand roughness which is obtained by

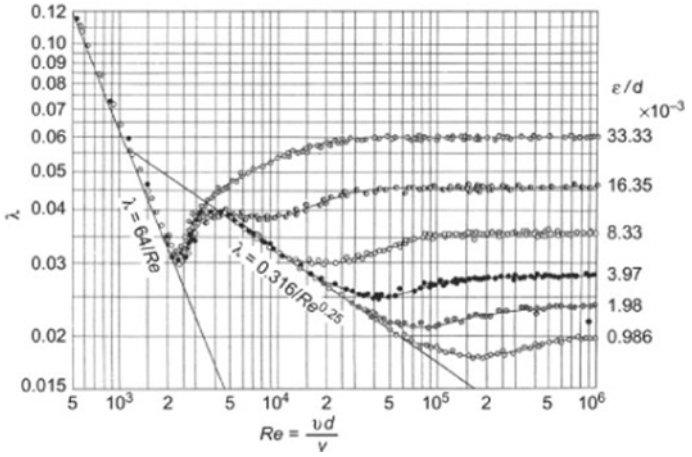


Fig. 2 Nikuradse's diagram. Adapted from [3]

fitting the same ΔU^+ to the Nikuradse's experiment. Therefore k_s is a hydrodynamic property rather a physical property of the roughness. As stated, detection of ΔU^+ is of significant role in roughness study.

In the past few decades, high fidelity simulations, especially Direct Numerical Simulations (DNS), have been increasingly used to predict ΔU^+ for a surface of interest—see e.g. [5] and the references therein. DNS is well known for its high accuracy of the result due to the absence of turbulence model i.e. the smallest turbulent motion is completely resolved in time and space. The computational cost of a conventional DNS scales with Re^3 [6]. Hence, in order to reach fully rough regime, high computational costs have to be taken into account.

In order to predict the roughness effect a priori, engineering correlations that relate the roughness geometry to flow quantities are required. A comprehensive roughness database is the key element to build such universal correlation. However, conventional way to construct roughness database by simulating realistic rough surfaces are inefficient; three major challenges here are,

- technical difficulties associated with scanning and digitization of realistic rough surfaces;
- limited number of available scanned surfaces leading to difficulties in estimation of roughness properties;
- large computational costs required for DNS.

With the focus on predicting ΔU^+ from rough surface of interest, computational domain can be reduced to fit the size of near-wall minimal flow unit according to the work by Chung et al. [7]. In the reduced computational domain, or called minimal channel, resolved turbulent length scale is limited by the size of the channel. Turbulence in the near wall flow can be completely resolved to simulate the interaction

of flow with roughness elements, thus the prediction of roughness function ΔU^+ remains accurate.

Scanning a realistic surface can be time consuming and expensive, so a systematic variation of roughness properties is technically difficult for a realistic surface. To overcome this limitation, a mathematical rough surface generation algorithm is proposed by Pérez-Ràfols et al. [8], in which Power Spectrum (PS) and Height Probability Distribution (HPD) of the rough surface can be prescribed. Correspondingly, the spatial distribution information and height distribution information of the rough surface are used as input parameters for the roughness generation. With the pseudo-random generation algorithm, roughness properties can be varied systematically and a comprehensive roughness database can thus be built.

In the present work technical routine of constructing roughness database using minimal channel simulation along with pseudo-random roughness generation algorithm is tested. A database with 12 rough surfaces with different properties is generated and the flow over the roughness is simulated. The employed numerical scheme and the HPC resources used for both minimal channels and full span channels are shortly described in the following sections. Scalability analysis for the utilized the pseudo-spectral solver SIMSON [9] is also carried out.

2 Numerical Procedure

Incompressible Navier-Stokes equations are solved using a pseudo-spectral solver SIMSON [9] on a Cartesian computational grid fine enough to allow full resolution of all flow scales down to the Kolmogorov scale as well as rough surface structures. A velocity-vorticity formulation is used with a spectral spatial and a semi-implicit temporal scheme. The computational domain is a box with periodic boundary conditions in the streamwise (x) and spanwise (z) directions. The sketch of simulation domain is shown in Fig. 3. In figure minimal channels, which will be explained in following text, as well as full span channel are shown, hatched pattern indicates roughness, roughness on the upper bound is not shown for simplicity. The wall-normal domain extension (y) is bounded by a no-slip boundary condition. The spatial scheme is based on Fourier and Chebyshev decompositions, in wall-parallel (x-z plane) and wall-normal (y) directions, respectively.

Roughness generation algorithm is realised by adjusting HPD and PS of the rough surface iteratively, the transformation between physical space and frequency space is done by Discrete Fast Fourier Transform (DFFT). An IBM based on the feedback-forcing method of Goldstein et al. [10] is used to enforce the zero-velocity (isothermal for passive scalar) condition in the solid sub-domain on the orthogonal computational grid. In addition, a constant source term representing the driving pressure gradient in the channel is also involved in the whole computational domain. The Navier-Stocks equations write:

$$\nabla \cdot \mathbf{u} = 0 \,, \tag{2}$$

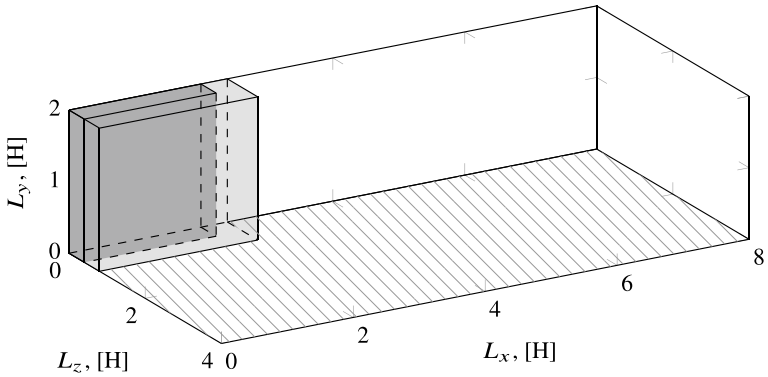


Fig. 3 Schematic of full-span channel (transparent) and minimal channels M1 (light gray) and M2 (deep gray), hatched pattern represents roughness. The rough surface on upper boundary is not shown

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{uu}) = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \mathbf{u} - \frac{1}{\rho} P_x \delta_{1i} + \mathbf{F}_u \ , \tag{3}$$

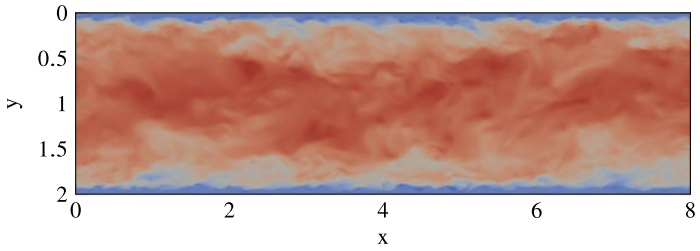
$$\frac{\partial \Theta}{\partial t} + \nabla \cdot (\mathbf{u}\Theta) = \alpha \nabla^2 \Theta + G + F_\Theta \ , \tag{4}$$

where $\mathbf{u}$ is the velocity vector $\mathbf{u} = (u, v, w)^\intercal$, P_x is the mean pressure gradient in the flow direction added as a constant and uniform source term to the momentum equation to drive the flow in the channel. Θ is a passive scalar represents the temperature. G is the source term in the flow. Here p, δ_{1i}, ρ, ν, α, $\mathbf{F}_u$ and F_Θ are pressure fluctuation, Kronecker delta, density, kinematic viscosity, heat transfer coefficient, external body force term due to IBM and the scalar source term for IBM, respectively. Figure 4 shows exemplary two instantaneous streamwise velocity fields. It is clear that flow field is affected by the different roughness geometry.

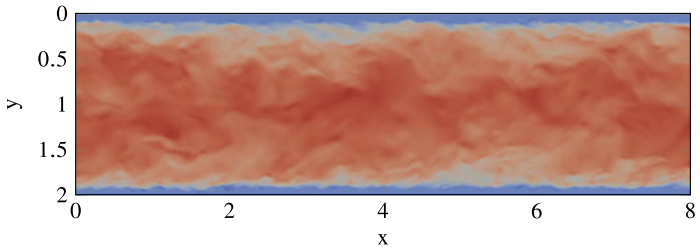
3 Construction of Roughness Database

3.1 Database Setup

Taking the advantage of roughness generation algorithm, rough surfaces will be generated based on systematically varied HPD and PS. Three properties of rough surfaces are reported to be relevant for the alteration of resultant skin friction in the literature [2, 11, 12]—skewness of HPD Sk, decline slope of PS p and the upper cutoff wavelength of the roughness wavelength λ_0. Minimal channel domain size is adjusted to resolve the largest roughness element. Two types of minimal channels are therefore introduced: $M1 = 2.4H \times 2H \times 0.8H$ corresponding to



(a) Positively skewed rough channel



(b)Negatively skewed rough channel

Fig. 4 Instantaneous streamwise velocity fields in x-z-plane issued from positively skewed roughness **a** and negatively skewed roughness **b**, both roughness share same power spectrum, flow direction from left to right

$\lambda_0 = 1.6H$; $M2 = 2.0H \times 2H \times 0.4H$ corresponding to $\lambda_0 = 0.8H$. Additionally for each roughness configuration a full span simulation ($8H \times 2H \times 4H$) is conducted for comparison. Summary of cases are listed in Table 1, abbreviations of each case are presented in the first column and λ_1 is the lower cutoff wavelength of roughness height function. After a mesh independence test, the streamwise and spanwise mesh resolution is selected with $\Delta x^+ < 5$ and $\Delta z^+ < 5$, respectively, while the wall-normal grid spacing is gradually stretched from wall to the middle of the domain using Chebyshev node distribution with 401 grid points.

3.2 Results

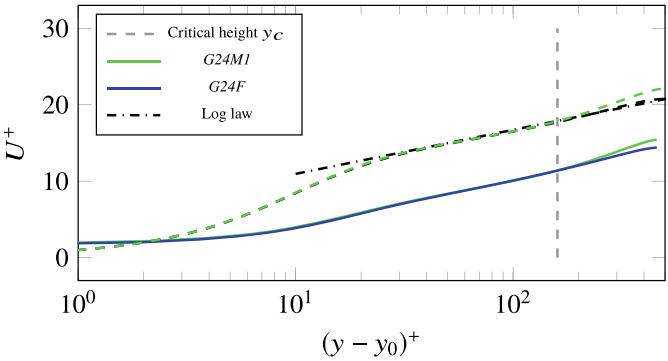
Mean velocity profile from both full-span channel and minimal channels corresponding to roughness configuration *G24* are illustrated in Fig. 5(a) and (b). Due to the limitation of channel size, predicted velocity profiles collapse from wall up until a certain height denoted as the critical height y_c which is reported $y_c \approx 0.4 \times L_z$ [7]. However, observing the velocity offset profile on the lower panel, profiles show consistence indicating that minimal channel simulation is capable for the prediction of roughness function ΔU^+. As is observed from Fig. 5(c) the velocity offset tend to be a constant in logarithmic layer. The roughness function ΔU^+ is defined as the

Table 1 Roughness properties of each cases

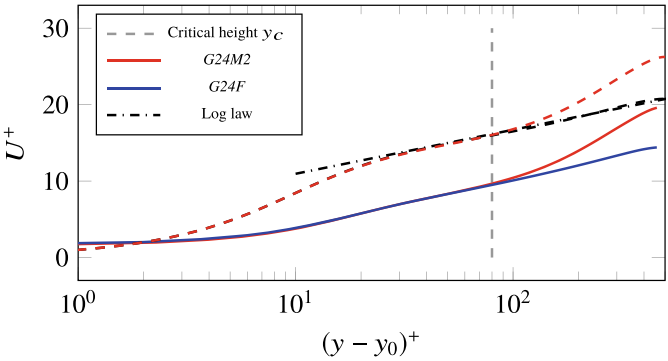
Case	*Sk*	*p*	$\lambda_0(H)$	$\lambda_1(H)$	Domain Size (H)
G14F	0	−1	0.8	0.08	8×4
G14M1	0	−1	0.8	0.08	2.4×0.8
G14M2	0	−1	0.8	0.08	2.0×0.4
G18F	0	−1	1.6	0.08	8×4
G18M1	0	−1	1.6	0.08	2.4×0.8
G24F	0	−2	0.8	0.08	8×4
G24M1	0	−2	0.8	0.08	2.4×0.8
G24M2	0	−2	0.8	0.08	2.0×0.4
G28F	0	−2	1.6	0.08	8×4
G28M1	0	−2	1.6	0.08	2.4×0.8
P14F	0.48	−1	0.8	0.08	8×4
P14M1	0.48	−1	0.8	0.08	2.4×0.8
P14M2	0.48	−1	0.8	0.08	2.0×0.4
P18F	0.48	−1	1.6	0.08	8×4
P18M1	0.48	−1	1.6	0.08	2.4×0.8
P24F	0.48	−2	0.8	0.08	8×4
P24M1	0.48	−2	0.8	0.08	2.4×0.8
P24M2	0.48	−2	0.8	0.08	2.0×0.4
P28F	0.48	−2	1.6	0.08	8×4
P28M1	0.48	−2	1.6	0.08	2.4×0.8
N14F	−0.48	−1	0.8	0.08	8×4
N14M1	−0.48	−1	0.8	0.08	2.4×0.8
N14M2	−0.48	−1	0.8	0.08	2.0×0.4
N18F	−0.48	−1	1.6	0.08	8×4
N18M1	−0.48	−1	1.6	0.08	2.4×0.8
N24F	−048	−2	0.8	0.04	8×4
N24M1	−0.48	−2	0.8	0.04	2.4×0.8
N24M2	−0.48	−2	0.8	0.04	2.0×0.4
N28F	−0.48	−2	1.6	0.04	8×4
N28M1	−0.48	−2	1.6	0.04	2.4×0.8

averaged velocity offset over logarithmic layer for the full-span channel, while for minimal channels, the function is defined as the offset at each critical height y_c.

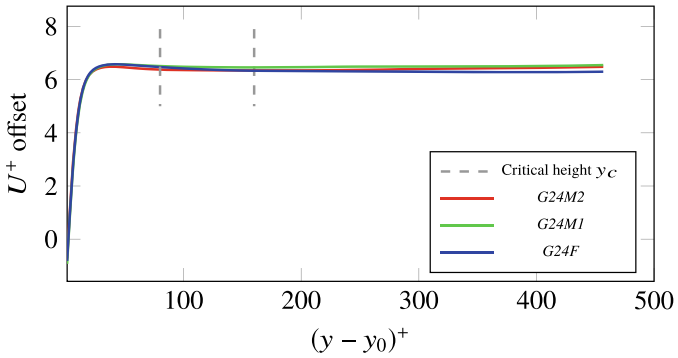
The Reynolds stresses from roughness topography $G24$ are visualized in Fig. 6. The location of characteristic roughness height $k^+ = 50$ is represented by vertical dashed line as a reference. From the graph reasonable agreement of Reynolds stresses between different channel sizes in the near wall region can be observed. Comparing full-span channels with M2 and M1, the peaks of $\overline{u'u'}$, $\overline{u'v'}$ and $\overline{v'v'}$ show good agreement, however, the profile of $\overline{w'w'}$ deteriorate from the full-span reference



(a) Mean velocity profile, full-span vs. *M1*



(b) Mean velocity profile, full-span vs. *M2*



(c) Velocity offset profile

Fig. 5 Mean velocity profiles and the roughness function for *G24*

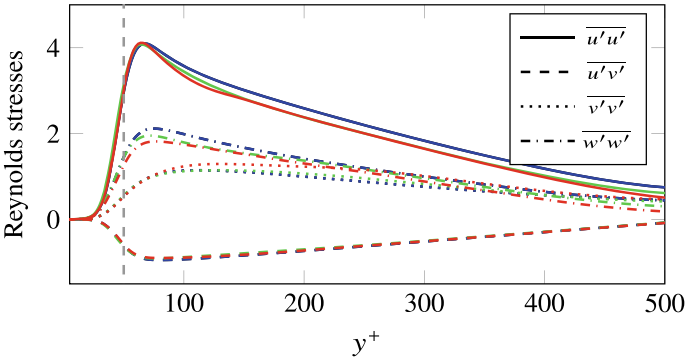


Fig. 6 Reynolds stresses from roughness topography $G24$, color indicates different channel sizes, blue: $G24F$, green: $G24M1$, red: $G24M2$

Table 2 Roughness function ΔU^+ estimated for minimal channels and full-span channels

Case	*F*	*M1*	*M2*	Case	*F*	*M1*	*M2*
G14	6.7	6.7	6.6	*G24*	6.3	6.4	6.4
G18	6.6	6.6	–	*G28*	5.9	5.9	–
P14	7.3	7.3	7.4	*P24*	7.0	6.9	6.9
P18	7.2	7.3	–	*P28*	6.6	6.9	–
N14	6.1	6.1	5.9	*N24*	5.8	5.6	5.8
N18	6.1	6.1	–	*N28*	5.5	5.6	–

startin from $y^+ = 50$. This could be linked to the direct effect of significantly reduced channel streamwise and spanwise extent which might lead to structural modification away from the wall and alteration of fluctuation intensity for $\overline{w'w'}$. Moving upwards to the middle part of the channel, due to the constraining effect of minimal channels, unphysical flow in outer layer might explain the slight discrepancy in the profiles. In spite of that, the capability of minimal channels in reproducing higher order statistics in the near wall region can be still considered as acceptable. Recalling the main goal of the project, in order to predict the roughness effect in engineering application, first order statistics are of crucial importance. Generalizing the analysis to all simulation cases, predicted roughness functions are listed in Table 2. Prediction error of the minimal channel relative to a full-span channel are illustrated in Fig. 7 against Skewness Sk. For all the cases, the estimated error is lower than 5%, also great consistence in the results is demonstrated. It is worth to mention that the surfaces in minimal channels as well as full-span channels are generated separately, i.e. each full-span/minimal channel pairs merely share identical topographical statistics since surface realizations are different.

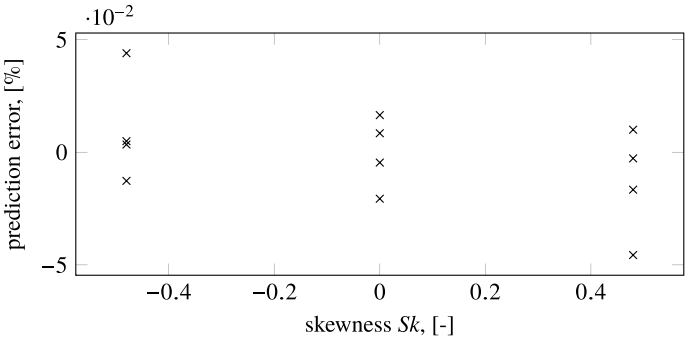


Fig. 7 Prediction error for minimal channels for different skewness

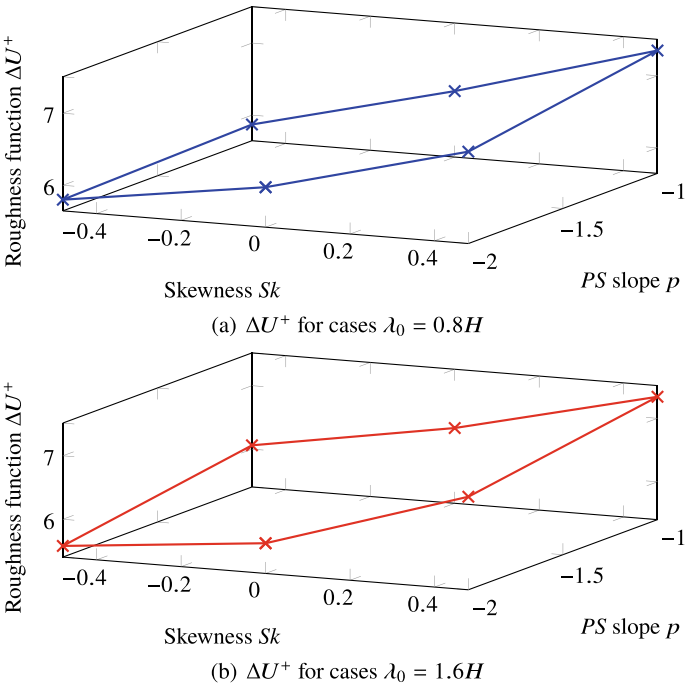


Fig. 8 Roughness function plots against Skewness Sk and decline slope p

Figure 8(a) shows the roughness function as a function of Sk and p with $\lambda_0 = 0.8H$ and Fig. 8 demonstrates the roughness function as a function of Sk and p while $\lambda_0 = 1.6H$. It can be observed that with increasing Sk roughness function ΔU^+ increases and a flatter PS leads to a higher roughness function values. Comparing ΔU^+ between the two plots, roughness function ΔU^+ decreases with increasing upper cutoff wavelength, indicating that those relative smaller roughness wavelengths

are dominant for the roughness function. The findings are found to be consistent with literature [2, 11].

4 HPC Aspects

4.1 Key Data

A total number of 40 simulations have been carried out at friction Reynolds numbers $Re_\tau = u_\tau \delta / \nu = 500$. In the following, HPC-related information of the typical simulations for a full-span channel and a minimal channel are presented:

- **full-span channel**
  - CPU hours per complete simulation: 200,000–300,000,
  - Computational grid size: $900 \times 401 \times 480$,
  - Maximum number of parallel cores: 900.
- **minimal channel**
  - CPU hours per complete simulation: 10,000–20,000,
  - Computational grid size: $256 \times 401 \times 48$,
  - Maximum number of parallel cores: 64.

The code is written in Fortran and parallelized in 2 dimensions using MPI. A good scalability is observed up to the largest number of cores used. It should be mentioned that each minimal channel simulation task takes only 10,000 to 20,000 CPU hours, while a full span DNS requires 200,000–300,000 CPU hours. Hence a considerable number of minimal channel simulations can be done instead simulataneously, which is ideal for the construction of massive database.

4.2 Scalability Analysis

The strong scalability of the code is tested on the hardware of ForHLR II cluster. The task size corresponds to the minimal channel with $L_x \times L_y \times L_z = 2.0H \times 2H \times 0.4H$ and $N_x \times N_y \times N_z = 256 \times 401 \times 48$ grid nodes. The size of the simulation tasks is kept constant and the number of cores employed is increased from 16 to 144. The simulation domain is divided in horizontal directions i.e. along streamwise and spanwise axis, so the side length of the sub-domain in each direction is identical. The simulation with 16 cores is taken as the reference to illustrate the speed up with the increased amount of cores.

The time per step and the speed-up are plotted in Fig. 9. The speed-up is defined:

Table 3 Strong scaling

Processors	Cells	Cells per Processor	Time each step	Speed up	CPU efficiency	Node efficiency
16	4.9×10^6	3.1×10^5	0.915 s	1	1	1
36	4.6×10^6	1.2×10^5	0.413 s	2.215	0.99	1.11
64	4.9×10^6	7.6×10^4	0.28 s	3.268	0.82	0.82
100	4.8×10^6	4.8×10^4	0.21 s	4.357	0.70	0.87
144	4.6×10^6	3.2×10^4	0.315 s	2.905	0.32	0.36

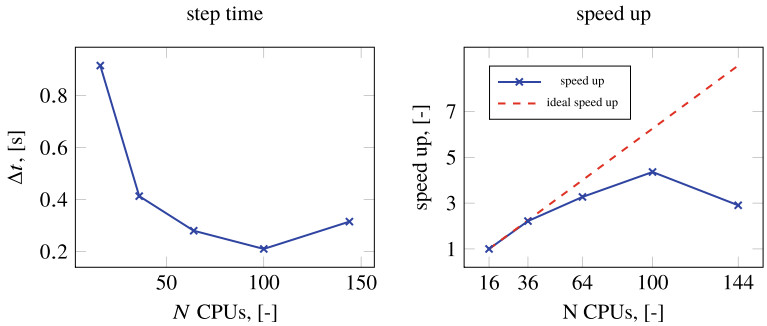


Fig. 9 CPU time for a single simulation time step (left panel) and speed-up (right panel)

$$S = \frac{\Delta_t}{\Delta_{t,ref}} \tag{5}$$

where Δt represents execution time of a single time step. The subscript ref indicates the case with 16 cores. One can observe that in order to acquire the best speedup, 100 cores should be activated. At the same time the node efficiency as well as the CPU efficiency are listed in Table 3 has to be taken into account. The CPU efficiency writes:

$$\eta_{CPU} = \frac{\Delta t \cdot N_{CPU}}{\Delta t_{ref} \cdot N_{CPU,ref}}, \tag{6}$$

where Δt and N_{CPU} represent execution time of a single time step and number of CPUs activated respectively. On the other hand, the node efficiency is given as:

$$\eta_{node} = \frac{\Delta t \cdot N_n}{\Delta t_{ref} \cdot N_{n,ref}}, \tag{7}$$

where N_n represent execution time of a single time step and number of nodes activated respectively, It is noticeable that the simulation with 36 cores gives a node efficiency greater than that of the reference case i.e. over unity. It should be mentioned that each computational node of ForHLR II has 20 cores, so the number of cores assigned to

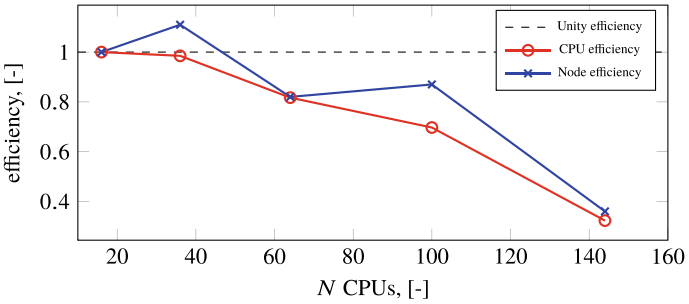


Fig. 10 Strong scaling efficiency

each node is kept identical in the study. Better utilization rate of CPUs is achieved in case with 2 nodes $\times$ 18 cores relative to case with 1 Node$\times$ 16 Cores. The CPU-based efficiency is plotted in Fig. 10. Although the case with 100 cores achieves the best utilization rate (5 nodes $\times$ 20 cores), due to the reduced overall CPU efficiency, the node efficiency for 100 cores significantly decreases. Based on the observation of speed-up and efficiency behavior for present simulation cases, 64 CPUs (4 nodes $\times$ 16 CPUs) is utilized for each minimal channel simulation.

5 Conclusion

Capability of minimal channel DNS is tested in present work, while pseudo-random roughness generation algorithm is employed to generate roughness surrogates. Both minimal channel simulations and full-span channel simulations are carried out showing consistent results and good agreement for estimation of the roughness function. After testing the framework of these two techniques, following findings are observed:

- minimal channel simulations can be used to predict roughness function within 5% deviation with significantly lower computational cost compared to full-span DNS,
- Reynolds stresses are successfully reproduced by minimal channels within the near-wall region,
- pseudo-random roughness generation algorithm shows great flexibility and stability and can be successfully employed for parametric study of rough surfaces,
- skewness of the roughness HPD can affect roughness function to a considerable extent,
- certain roughness wavelengths are identified to be dominant for the roughness function,
- significant progress in studying rough surfaces in turbulent flows can achieved by employing the technical framework of the present project.

References

1. R. Webb, E. Eckert, R. Goldstein, Heat transfer and friction in tubes with repeated-rib roughness. Int. J. Heat Mass Transf. **14**(4), 601–617 (1971)
2. K. Flack, M. Schultz, J. Barros, Skin friction measurements of systematically-varied roughness: Probing the role of roughness amplitude and skewness. Flow Turbul. Combust. **104**, 317–329 (2019)
3. J. Nikuradse, *Strömungsgesetze in Rauhen Rohren* (VDI-Verl., Berlin, 1933). (ISBN 0341-7247)
4. A. Townsend, *The Structure of Turbulent Shear Flow*. (Cambridge University Press, Cambridge, 1980). (ISBN 052120710). http://www.loc.gov/catdir/toc/cam032/74014441.html
5. P. Forooghi, A. Stroh, F. Magagnato, S. Jakirlic, B. Frohnapfel, Toward a universal roughness correlation. ASME J. Fluids Eng. **139**, 121201 (2017)
6. S.B. Pope, *Turbulent Flows* (Cambridge University Press, Cambridge, 2000)
7. D. Chung, L. Chan, M. MacDonald, N. Hutchins, A. Ooi, A fast direct numerical simulation method for characterising hydraulic roughness. J. Phys. Conf. Ser. **708**, 012010 (2016)
8. F. Pérez-Ràfols, A. Almqvist, Generating randomly rough surfaces with given height probability distribution and power spectrum. Tribol. Int. **131**, 591–604 (2019)
9. M. Chevalier, P. Schlatter, A. Lundbladh, D.S. Henningson, SIMSON-a pseudo-spectral solver for incompressible boundary layer flow. Technical report TRITA-MEK, pp. 1–100 (2007)
10. D. Goldstein, R. Handler, L. Sirovich, Modeling a no-slip flow boundary with an external force field. J. Comput. Phys. **105**(2), 354–366 (1993)
11. J. Barros, M. Schultz, K. Flack, Measurements of skin friction of systematically generated surface roughness. Int. J. Heat Fluid Flow **72**, 1–7 (2018)
12. K. Flack et al., Skin-friction behavior in the transitionally-rough regime. Int. J. Heat Fluid Flow **61**, 21–30 (2016)

Studies on the Applicability of a Simplified Gust Simulation Approach in the CFD Code TAU

Jens Müller, Marco Hillebrand, Maximilian Ehrle, Michael Schollenberger, Thorsten Lutz, and Ewald Krämer

Abstract A simplified method for gust modeling in CFD simulations, the so-called Disturbance Velocity Approach (DVA), is analyzed within this work. This approach covers the influence of a gust on an aircraft but neglects the influence of the aircraft on the gust. Simulations of an airfoil encountering different "1-cos" gusts were performed. The results of the DVA are compared to simulations where the gust is fully resolved and propagated within the flow field and all interactions are considered. It is shown that the simplified approach provides satisfying results for gust wavelengths larger than the chord length. At smaller wavelength significant deviations to the resolved simulations exist especially regarding the pitching moment. In the present study angle of attack and airfoil shape were varied and the aerodynamic effects influencing the accuracy of the simplified approach are identified. The DVA also provides satisfying results for superposition of different gust signals. A scaling test of the TAU code on the HPE Hawk demonstrates good scalability.

1 Introduction

Aircraft encountering atmospheric gusts experience variations in the local angle of attack. This leads to fluctuations of the aerodynamic forces and, as a result, increased structural loads. Therefore, a detailed assessment of gust loads on aircraft has to be carried out already in the early phases of aircraft design. These loads also have to be considered in the certification process [1]. Hence, a fast and cost-efficient method that enables industrial application is required. Currently, different methods exist in order to calculate the response of an aircraft to atmospheric gusts. Models based on linear potential flow theory like the doublet-lattice or the unsteady vortex lattice method [2] are widely used in industry due to their fast computation time and low costs. While these methods neglect transonic and viscous effects, CFD simulation provides a more accurate approach. Different methods are available to represent

J. Müller (✉) · M. Hillebrand · M. Ehrle · M. Schollenberger · T. Lutz · E. Krämer
Faculty 6: Aerospace Engineering and Geodesy, Institute of Aerodynamics and Gas Dynamics, University of Stuttgart, Pfaffenwaldring 21, 70569 Stuttgart, Germany
e-mail: jens.mueller@iag.uni-stuttgart.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_26

atmospheric gusts in CFD simulations, see also [2] and [3]. The most accurate is the Resolved Atmosphere Approach (RAA), where the gust is fed into the flow field at the far field boundary and is then resolved and propagated within the flow field. This methods requires a very fine mesh resolution in order to avoid any numerical losses during the propagation of the gust from the far field boundary to the aircraft. Hence, this method is very demanding in terms of computational resources but covers both the interaction of the gusts with the aircraft as well as the effect of the aircraft aerodynamics on the gust [4]. In order to reduce the computational effort simplified methods like the Disturbance Velocity Approach (DVA) [4], also called field velocity method, or the split velocity method [5] were developed. The present work focuses on the DVA.

The DVA adds the velocities of atmospheric disturbances like gusts to the flux balance by superposition. This enables the usage of non-refined standard grid resolutions which reduces the computational effort significantly. The main drawback is that only the influence of the gust on the airfoil but not the influence of the airfoil on the gust is captured. The DVA was implemented into the flow solver TAU [6] by Heinrich and Reimer [4]. They analyzed the differences between the simplified DVA and the RAA in terms of lift. Their studies involve a two dimensional NACA 0012 airfoil with a Horizontal Tail Plane (HTP) and a realistic aircraft configuration encountering vertical "1-cos" gusts of different wavelengths [4, 7]. They conclude that the DVA provides reasonable results for gust wavelengths larger than the chord length. The applicability of the DVA to broadband atmospheric turbulence interacting with an extruded airfoil and a swept wing is analyzed in [8] and [9] where the DVA provides satisfying results compared to the RAA.

Since the gust interaction studies in [4] and [7] involve two element or complex configurations and focus only on the lift response, the scope of the work at hand is to evaluate and identify the aerodynamic effects influencing the accuracy of the DVA in more detail. Therefore, CFD simulations of an airfoil encountering vertical "1-cos" gusts are performed and the responses of lift, drag, and pitching moment are evaluated. Results of the simplified DVA are compared to the more accurate RAA simulations. Gust wavelength, angle of attack, and airfoil shape are varied to identify the main aerodynamic effects influencing the accuracy of the DVA. Additionally, the applicability of the DVA to a gust signal consisting of different superimposed wavelengths is analyzed. Finally, the performance of the DLR TAU code on the high performance computing system HPE Hawk is evaluated.

2 Numerical Methods and Setup

The unstructured Navier-Stokes solver TAU [6] developed by the German Aerospace Center (DLR) is used for the simulations within this work. TAU is of second order accuracy in space and time and is based on the finite volume approach. As mentioned in Sect. 1, two different methods, the Resolved Atmosphere Approach (RAA) and the Disturbance Velocity Approach (DVA) are used for the CFD simulations of gust

interaction within this work. Both methods as well as the requirements for gust simulations in terms of mesh resolution and simulation procedure are described in detail in [4] and [7] and are only briefly introduced here.

The RAA represents a physical correct method to represent gust interaction in CFD simulation. Here, the gust is fed into the flow field at the far field boundary. Therefore, the standard far field boundary condition is modified to allow a transient velocity specification. Instead of setting a constant inflow velocity $\mathbf{u}_\infty$ at all spatial positions of the far field boundary condition at any time, the velocity components of the gust can be specified as function of space and time [4]. Hence, the velocity specification at the modified boundary condition changes from $\mathbf{u}_\infty$ to $\mathbf{u}_\infty + \mathbf{u'(t)}$ where $\mathbf{u'(t)}$ is the time dependent gust velocity. The gust is then resolved and propagated within the discretized flow field. Within the DVA the time dependent disturbance velocity vector $\mathbf{u}'$ of the gust is not fed into the flow field but added to the flux balance by superposition. Regarding the continuity equation this leads to

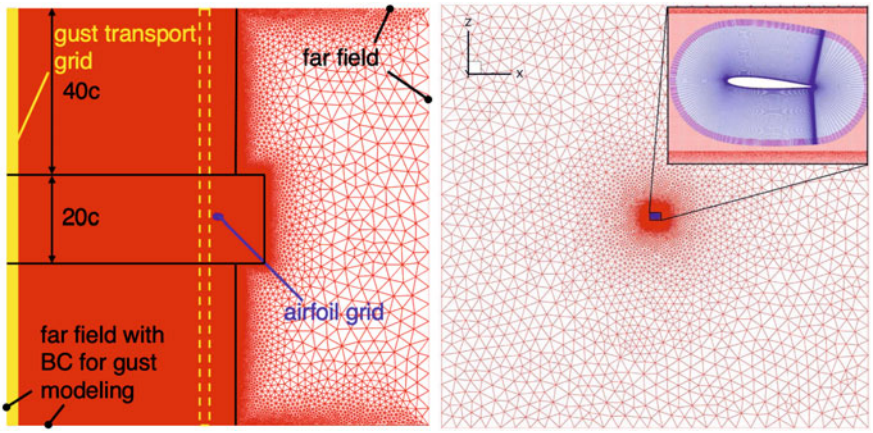
$$\frac{d}{dt}\int_V \rho \, dV + \oint_S \rho(\mathbf{u} - \mathbf{u}_b - \mathbf{u}') \cdot \mathbf{n} \, dS = 0 \tag{1}$$

where the convection across the cell interfaces is changed from $\mathbf{u} - \mathbf{u}_b$ to $\mathbf{u} - \mathbf{u}_b - \mathbf{u}'$. $\mathbf{u}_b$ is the velocity of the boundary of a control volume. Hence, the gust velocity acts similar to a grid velocity. This enables the use of non-refined standard CFD grid resolutions which allows an industrial application. The main drawback is that the velocities and the shape of the gust remains unchanged during the entire simulation. Thus, only the influence of the gust on the airfoil is captured and the influence of the interaction with the airfoil on the gust is neglected. This leads to deviations in the gust interaction results of the DVA compared to the RAA where all physical effects are covered.

All simulations within this work involve an airfoil encountering a vertical gust. The inflow conditions are subsonic with $M = 0.25$ and $Re = 11.6 \cdot 10^6$. According to the EASA certification specification CS 25.341 [1] the vertical velocity w of a vertical "1-cos" gust is

$$w(x) = \frac{A_{gust}}{2}\left[1 - \cos\left(\frac{2\pi x}{\lambda}\right)\right] \quad \text{with} \quad 0 \leq x \leq \lambda \tag{2}$$

with gust amplitude A_{gust} and wavelength λ. Based on the gust analysis of Heinrich and Reimer in [4] and [7] a gust amplitude of 10% relative to the inflow velocity u_∞ is chosen. To consider different angles of attack in the RAA and DVA calculations of a "1-cos" gust, the airfoil grid has to be separated from the background grid. Therefore, a structured airfoil grid is generated and integrated into a background grid using the Chimera method. The angle of attack is imposed by a rotation of the airfoil grid around the quarter chord. The airfoil grid consist of a boundary layer grid based on three parts, the upper and lower surface with 150 points each and the trailing edge with 35 points, and a fully structured near field grid. $y^+ \leq 1$ is ensured



a) RAA grid b) DVA grid

Fig. 1 RAA and DVA grid setup with background grid (red), airfoil grid (blue), and gust transport grid (yellow). The dashed yellow line represents the final position of the gust transport grid.

and the growth rate in wall normal direction is set to 1.18. The airfoil grid is similar for both RAA and DVA simulations and highlighted in blue in Fig. 1b. The entire flow field and thus the background grid extends over an area of $100c \times 100c$ with c being the chord length of the airfoil. Figure 1 shows the grids of the RAA and DVA including the background grid, the airfoil grid, and the gust transport grid, where the latter is only used for the RAA.

As mentioned above, when using the RAA the gust has to be propagated from the far field boundary $50c$ upstream from the airfoil's leading edge to the airfoil without significant numerical losses. This requires a very fine grid resolution in the propagation region, especially when considering the second order spatial accuracy of TAU. To reduce the amount of grid cells an overset gust transport grid based on the Chimera technique is used as proposed by Heinrich and Reimer in [4] for RAA gust simulations in TAU. The gust transport grid has a width of 2λ in streamwise direction. Its grid resolution $\Delta x = \Delta z = 0.01c$ is based on the density study of the gust transport grid in [4] with $\lambda = c$ being the smallest gust wavelength considered within this work. The gust is fed in at the far field boundary condition with modification for gust modeling on the left and lower side of the computational domain as illustrated in Fig. 1a. At the start of the simulation the transport grid is located on the left side of the computational domain as indicated in Fig. 1a. When the gust is located in the center of the transport grid, the transport grid starts to move with the convection velocity u_∞ of the flow. This ensures that there are no numerical losses during the propagation to the airfoil since gust transport grid and flow move with the same speed. The transport grid stops close to the airfoil grid with the final position indicated by the dashed yellow lines in Fig. 1a. Then the gust leaves the transport grid and interacts with the airfoil. The background grid of the RAA consist of a structured and unstructured domain. The structured domain includes two domains of

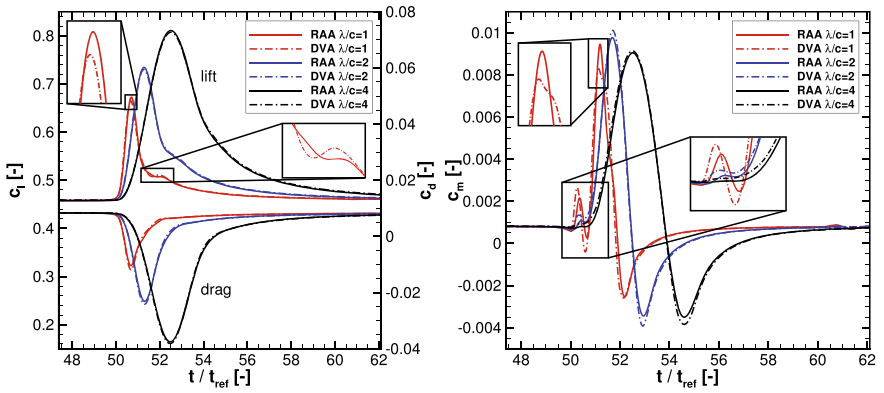
different grid resolution. A high resolution of $0.01c$ in x- and z-direction is used for the domain in the middle of the grid to avoid accuracy errors during the gust transport. This domain has a height of $20c$ and starts at the inflow boundary upstream from the airfoil and ends at a distance of $11c$ downstream from the airfoil's leading edge. The second structured domain extends $40c$ connecting the borders of the fine domain with the far field boundary. Thereby, the grid resolution in x-direction remains constant while the resolution in z-direction is reduced. This domain ends $5c$ downstream from the leading edge. The remaining flow field downstream from the structured grid is realized with an unstructured grid. The DVA background grid is unstructured with a structured part in the vicinity of the airfoil grid. The structured part has a resolution of $0.01c$ in x- and z-direction to avoid any differences regarding the Chimera interpolation between background and airfoil grid. Since the DVA does not require any specific grid refinement the grid resolution is coarsened in the unstructured part with increasing distance from the airfoil. All simulations use the unsteady RANS equations with the Spalart-Allmaras turbulence model applied. A physical time step of $\Delta t/t_{ref} = 0.005$, where $t_{ref} = c/u_\infty$, with 250 inner iterations per time step is applied for both RAA and DVA simulations.

3 Results

In order to enable a detailed evaluation of the applicability of the DVA to gust interaction simulation different aerodynamic parameters are varied for both, RAA and DVA simulation. First, the effect of different gust wavelengths on a NACA 0012 airfoil at moderate angle of attack is presented in Sect. 3.1 and the critical wavelength with the largest differences between DVA and RAA is identified. Subsequently, the angle of attack and the airfoil shape are varied for the critical wavelength. These results are presented in Sect. 3.2 and 3.3 respectively. In Sect. 3.4 the superposition of separate gust simulations compared to a single simulation using a superimposed gust is analyzed for the RAA and the DVA.

3.1 Variation of Gust Wavelength

The gust analysis of Heinrich and Reimer in [4] and [7] show that the deviations of the DVA lift signal compared to the RAA increase the smaller the gust wavelength λ compared to the chord length. Their studies utilized a 2D simulation of a symmetrical NACA 0012 airfoil with horizontal tail plane at zero angle of attack and a 3D simulation of an aircraft configuration. Hence, the effect of three different gust wavelengths, $\lambda/c = 1, \lambda/c = 2, \lambda/c = 4$ on a single NACA 0012 profile at $\alpha = 4°$ is analyzed in the work at hand using RAA and DVA. Similar to [4] the gust amplitude A_{gust} in Eq. (2) is set to 10% of the inflow velocity u_∞. In addition to the lift signal analyzed by Heinrich and Reimer, the drag and the pitching moment are considered



a) lift, drag b) pitching moment

Fig. 2 Time history of lift, drag, and pitching moment of a NACA 0012 airfoil at $\alpha = 4°$ encountering "1-cos" gusts of different wavelengths.

as well. This enables a detailed evaluation of the applicability of the DVA and an identification of the aerodynamic effects leading to the deviations between RAA and DVA. Figure 2 shows the time history of lift coefficient c_l, drag coefficient c_d, and pitching moment coefficient c_m around the quarter chord of the airfoil for all wavelengths considered. Here, and in all subsequent plots, the time is normalized by the reference time $t_{ref} = c/u_\infty$, the time a gust needs to travel a distance of one chord length. In all simulations the gust reaches the airfoil at $t/t_{ref} = 50$.

The DVA lift signal agrees well with the RAA lift signal for $\lambda/c = 2$ and $\lambda/c = 4$. The maximum lift is slightly smaller for the DVA compared to the RAA. This deviation of the maximum lift is more significant for $\lambda/c = 1$. At this wavelength an additional local c_l oscillation occurs in the DVA simulation between $51.4 \leq t/t_{ref} \leq 52.3$ when the gust is located only at the rear part of the airfoil or downstream from it. This oscillation, at the time the gust leaves the airfoil, is less pronounced for the RAA simulation as visible in the enlarged part of Fig. 2a. A list of the differences between RAA and DVA at the extreme points is provided in Table 1 with

$$err_{c_{l,max}} = \frac{|c_{l,max,RAA} - c_{l,max,DVA}|}{|c_{l,max,RAA}|} \quad \text{and} \quad \Delta c_i = c_{i,RAA} - c_{i,DVA}. \tag{3}$$

The relative c_l error is 1.18% for the smallest wavelength and well below 1% for the other wavelengths. Δc_l also decreases with increasing λ/c.

The shape of the drag coefficient time history is comparable between RAA and DVA for all wavelengths. The drag extremum is overestimated by the DVA compared to the RAA. In line with the lift signal, the deviation between RAA and DVA decreases with increasing λ/c as visible in Fig. 2a and Table 1. The negative drag occurs since the aerodynamic coordinate system used for the lift and drag evaluation is aligned with the inflow velocity u_∞ and the global angle of attack at the far field boundary

Table 1 Deviations between the RAA and DVA results at the extreme points of the global load response for different gust wavelengths.

	$\Delta c_{l,max}$	$\text{err}_{c_{l,max}}$	$\Delta c_{d,min}$	$\Delta c_{m,max}$	$\frac{\Delta t_{c_l}}{t_{ref}}$	$\frac{\Delta t_{c_d}}{t_{ref}}$	$\frac{\Delta t_{c_{m,max}}}{t_{ref}}$
$\lambda/c = 1$	0.00796	1.18%	0.001498	0.001166	0.035	−0.005	0.055
$\lambda/c = 2$	0.00453	0.62%	0.001023	−0.000339	0.005	0.010	−0.010
$\lambda/c = 4$	0.00365	0.45%	0.000594	−0.000095	0.000	0.005	0.010

and not with the local angle of attack, which changes when the gust interacts with the airfoil.

The RAA pitching moment around the quarter chord for $\lambda/c = 1$ shown in Fig. 2b has four turning points. When the gust front starts to interact with the airfoil the suction peak increases while the pressure distribution at the rear part of the airfoil is nearly unaffected. This leads to the local maximum at $t/t_{ref} = 50.36$. Hence, the center of the gust is located $0.14c$ upstream from the leading edge. With increasing downstream position of the gust the suction peak increases, but also the pressure distribution at the rear part of the airfoil is more affected, especially on the lower side. This leads to increased lift at the rear part of the airfoil that results in the local minimum at $t/t_{ref} = 50.67$. The gust maximum is located at $0.17c$. This is close to the point of maximum lift which occurs at $t/t_{ref} = 50.7$. At $t/t_{ref} = 51.19$ the global pitching moment maximum is reached. With the gust center located at $0.69c$ this is after the point of maximum lift. Both suction peak and lift of the rear part of the airfoil decrease but the influence of lift decrease at the rear part of the airfoil on the pitching moment is higher due to the larger distance to the quarter chord. When the center of the gust is located at $1.69c$ at $t/t_{ref} = 52.19$, the global minimum of the pitching moment is reached. Since the gust is located just downstream from the trailing edge, it mainly influences the trailing edge region resulting in a negative pitching moment. With increasing λ/c ratio the two local extreme points diminish due to the smaller velocity gradients of the gust and the increased wavelength. When comparing RAA and DVA results the differences are more significant than for c_l and c_d, especially for $\lambda/c = 1$. The local maximum and minimum at $\lambda/c = 1$ is overestimated by the DVA compared to the RAA whereas the global maximum is smaller. The absolute values of the global minimum and the extreme points for $\lambda/c = 2$ and $\lambda/c = 4$ are higher for the DVA results. The deviations between RAA and DVA at the maximum of the global pitching moment decrease with increasing λ/c, as illustrated in Table 1. Before the global maximum is reached there is a significant time shift present in the c_m signal, especially for $\lambda/c = 1$. The turning points are reached earlier for the DVA simulation. The main reason for this time shift is that the gust shape changes due to the interaction with the airfoil which is covered by the RAA and not covered by the DVA where the gust remains unchanged during the simulation. This variation of the gust shape in the RAA simulation is illustrated in Fig. 3.

Due to the circulation around the airfoil caused by the angle of attack the flow above the airfoil is accelerated while it is decelerated below the airfoil. This leads to

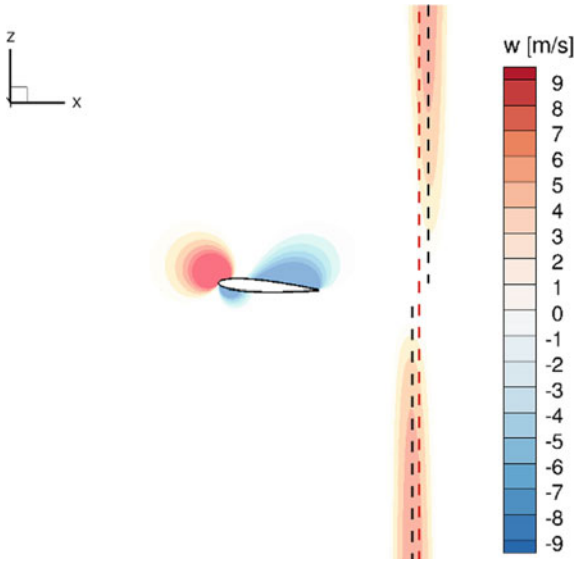


Fig. 3 Gust shape variation in the RAA simulation of the NACA 0012 airfoil at $\alpha = 4°$ with $\lambda/c = 1$. The dashed black lines indicate the center of the gust above and below the airfoil for the RAA, whereas the dashed red line indicates the center of the analytical gust position used by the DVA. The center of the analytical gust is located $2c$ downstream from the leading edge.

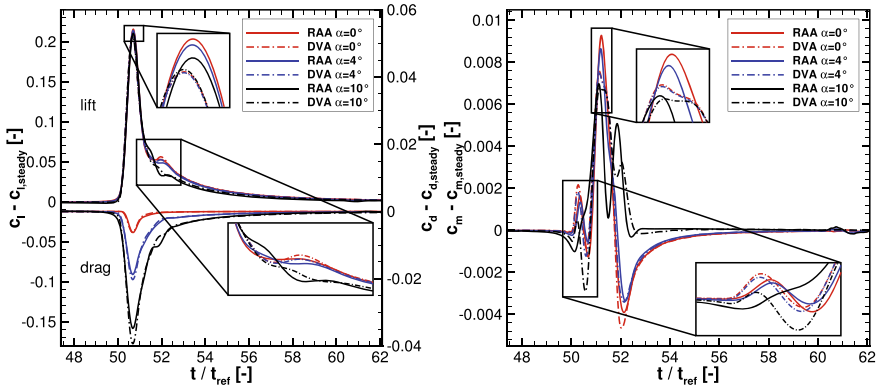
an offset of the gust position above and below the airfoil as indicated by the dashed black lines in Fig. 3. The analytical gust position as utilized by the DVA is indicated by the dashed red line. The spatial difference in streamwise direction between the gust maximum above and below the airfoil in the RAA simulation is denoted as Δx. Besides the offset in gust position the vertical velocity component of the gust is decreased in the vicinity of the airfoil also covered by the RAA but not by the DVA. These effects reduce the impact of the gust on the pitching moment resulting in the less pronounced turning points in Fig. 2. The offset in gust position is also the main reason for the time difference between the RAA and the DVA signal. With increasing wavelength the offset in gust position is of less importance since the gust velocity gradients decrease and the area of influence of the gust increases. Thus, the time difference Δt between RAA and DVA decreases with increasing λ/c for both, c_l and c_m as shown in Table 1.

Overall, the results of RAA and DVA at higher wavelengths agree well despite small deviations in the time history of the pitching moment. The smallest wavelength $\lambda/c = 1$ is the most critical when evaluating the applicability of the simplified DVA. This corresponds with the results in [4] for the airfoil/HTP configuration. Hence, $\lambda/c = 1$ is considered for the studies of the angle of attack and airfoil variations in the subsequent sections.

3.2 *Angle of Attack Variation*

Having identified the critical wavelength for the DVA to be accurate, the influence of different angles of attack on the NACA 0012 airfoil interacting with a vertical "1-cos" gust with a gust amplitude of 10% u_∞ and $\lambda/c = 1$ is evaluated within this section. Three different angles of attack are considered. $\alpha = 4°$ already presented in Sect. 3.1, $\alpha = 0°$, and $\alpha = 10°$. Since the NACA 0012 airfoil is symmetric, no difference in the gust position above and below the airfoil occurs in the RAA simulation at $\alpha = 0°$ whereas the acceleration above the airfoil and the deceleration below the airfoil increases with increasing α. Hence, the relative position Δx between the gust above and below the airfoil increases accordingly. In addition, higher velocity gradients exist at the leading edge region due to the more pronounced suction peak at higher angles of attack which can influence the shape of the gust within the RAA simulation. Hence, the angle of attack variation enables a detailed evaluation of these effects regarding the applicability of the DVA compared to the RAA. According to Sect. 3.1, c_l, c_d, and c_m at all three angles of attack are plotted in Fig. 4. In order to separate the influence of the gust from the steady differences in the force and moment coefficients due to the different angles of attack, the difference of the coefficients to the steady solution is plotted in Fig. 4 instead of the absolute values shown in Fig. 2. The differences between RAA and DVA at the global extreme points are presented in Table 2.

As for $\alpha = 4°$ the differences in the lift history of RAA and DVA shown in Fig. 4a are small. At all α the maximum lift is higher for the RAA than for the DVA simulation. With increasing α the relative c_l difference at the maximum lift of the DVA decreases from 4.21% at $\alpha = 0°$ to 0.24% at $\alpha = 10°$. The time difference of maximum c_l between RAA and DVA seems to be independent of the angle of attack



a) lift, drag b) pitching moment

Fig. 4 Time history of lift, drag, and pitching moment of a NACA 0012 airfoil at $\alpha = 0°$, $\alpha = 4°$, and $\alpha = 10°$ encountering a "1-cos" gust with $\lambda/c = 1$.

Table 2 Deviations between the RAA and DVA results at the extreme points of the global load response for different angles of attack.

	Δc_l	err_{c_l}	Δc_d	$\Delta c_{m,max}$	$\frac{\Delta t_{c_l}}{t_{ref}}$	$\frac{\Delta t_{c_d}}{t_{ref}}$	$\frac{\Delta t_{c_{m,max}}}{t_{ref}}$
$\alpha = 0°$	0.00908	4.21%	0.000032	0.001694	0.030	0.020	0.090
$\alpha = 4°$	0.00796	1.18%	0.001498	0.001166	0.035	−0.005	0.055
$\alpha = 10°$	0.00315	0.24%	0.004541	0.000168	0.035	0.005	−0.045

variation, see Table 2. Hence, the relative deviations Δx of the RAA gust position above and below the airfoil as well as the deviations to the analytical DVA gust position, which has no variations above and below the airfoil, have no negative effect on the maximum lift accuracy of the DVA. This is not the case when the gust leaves the airfoil for $51.4 \leq t/t_{ref} \leq 52.3$. While the course of the local c_l fluctuations of RAA and DVA shows only minor deviations at $\alpha = 0°$ and $\alpha = 4°$, more significant differences in the c_l course are visible at $\alpha = 10°$. At higher angles of attack and with increasing dowstream position of the gust, the relative difference of the RAA gust position above and below the airfoil increases. When the gust leaves the airfoil this effect becomes more significant. Therefore, the DVA time history of the c_l signal at $\alpha = 10°$ differs from the RAA signal when the gust leaves the airfoil.

Regarding the drag coefficient, the agreement between RAA and DVA decreases with increasing α, as illustrated in Fig. 4a and Table 2. As for c_l significant differences occur at $\alpha = 10°$ when the gust leaves the airfoil due to the difference of the relative gust position in the RAA simulation not covered by the DVA. In contrast to the lift coefficient the DVA results in higher absolute values of the c_d extremum with increasing difference between DVA and RAA at higher angles of attack.

When looking at the pitching moment in Fig. 4b the differences in the shape of the c_m time history of RAA and DVA at $\alpha = 0°$ and $\alpha = 4°$ are comparable. Considering the absolute differences between RAA and DVA, they are smaller for $\alpha = 4°$ than for $\alpha = 0°$, see also Table 2. This also applies to the time difference of the maximum pitching moment. At $\alpha = 10°$ the velocity gradients in the vicinity of the leading edge are higher than for the lower angle of attacks due to the more pronounced suction peak. This directly influences the gust shape within the RAA simulation when the gust interacts with the leading edge. The velocity reduction in the vicinity of the airfoil, already described and illustrated in Sect. 3.1 and Fig. 3, is more pronounced for $\alpha = 10°$ than for $\alpha = 4°$. As mentioned above, the relative difference of the gust position below and above the airfoil is increased as well. These effects have a distinct influence on the pitching moment which is covered by the RAA but not by the DVA. This results in clearly visible differences between RAA and DVA especially at the time the gust starts to interact with the leading edge and when the gust is located just downstream from the trailing edge. However, the global c_m maximum of RAA and DVA at $\alpha = 10°$ shows better agreement than at lower angle of attacks.

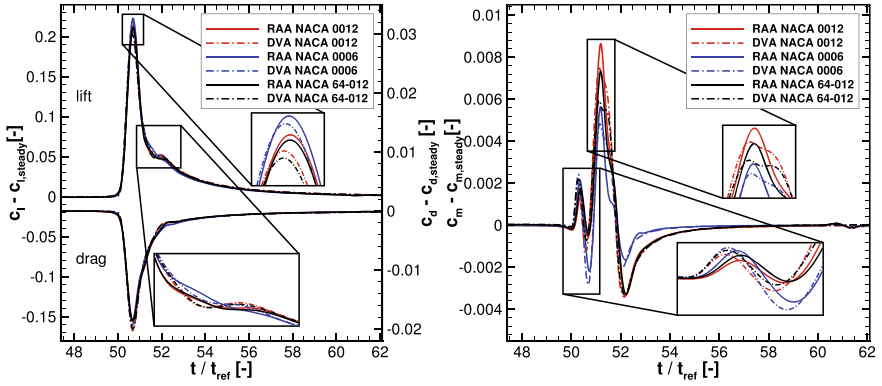
Consequently, the angle of attack has a visible influence on the accuracy of the DVA compared to the RAA. At high angles of attack the DVA pitching moment shows

considerable differences to the RAA but also the agreement in c_d is decreased. On the other hand the maximum lift and maximum pitching moment are well predicted by the DVA at $\alpha = 10°$.

3.3 *Variation of Airfoil Shape*

Besides wavelength and angle of attack, the shape of the airfoil can also affect the accuracy of the DVA. Therefore, three different airfoils encountering a vertical "1-cos" gust are compared when using RAA and DVA. The gust is similar to Sect. 3.2 with $A_{gust} = 0.1u_\infty$ and $\lambda/c = 1$. The angle of attack is set to 4°. The NACA 0012 airfoil presented in Sect. 3.1 and 3.2 is used as a reference. Two different variations of the airfoil shape are analyzed. The first is a reduction of the maximum thickness of the airfoil from 12%c to 6%c by using a NACA 0006 airfoil instead of the NACA 0012. This leads to higher pressure gradients at the leading edge region and a significantly more pronounced suction peak. The pressure gradients on the NACA 0006 airfoil downstream from the suction peak are reduced compared to the NACA 0012 airfoil. As second variation a NACA 64-012 airfoil is used. It has a different surface pressure characteristic and the point of maximum thickness is shifted from 30%c for the NACA 0012 airfoil to 40%c. The temporal evolution of c_l, c_d, and c_m is plotted in Fig. 5. As for the α variation presented in Sect. 3.2, only the difference to the steady solution is plotted in this figure. Table 3 lists the differences between RAA and DVA at the global extreme points.

The stronger surface pressure gradients of the NACA 0006 airfoil at the suction peak lead to an increased difference Δx of the gust maximum above and below the airfoil in the RAA simulation compared to the NACA 0012 in the vicinity of the



a) lift, drag b) pitching moment

Fig. 5 Time history of lift, drag, and pitching moment of a NACA 0012, a NACA 0006, and a NACA 64-012 airfoil at $\alpha = 4°$ encountering a "1-cos" gust with $\lambda/c = 1$.

Table 3 Deviations between the RAA and DVA results at the extreme points of the global load response for different NACA airfoils.

Profile	Δc_l	err_{c_l}	Δc_d	$\Delta c_{m,max}$	$\frac{\Delta t_{c_l}}{t_{ref}}$	$\frac{\Delta t_{c_d}}{t_{ref}}$	$\frac{\Delta t_{c_{m,max}}}{t_{ref}}$
NACA 0006	0.00368	0.54%	0.000743	0.000805	0.015	0.000	0.025
NACA 0012	0.00796	1.18%	0.001498	0.001166	0.035	−0.005	0.055
NACA 64-012	0.00886	1.34%	0.001404	0.001383	0.030	−0.005	0.065

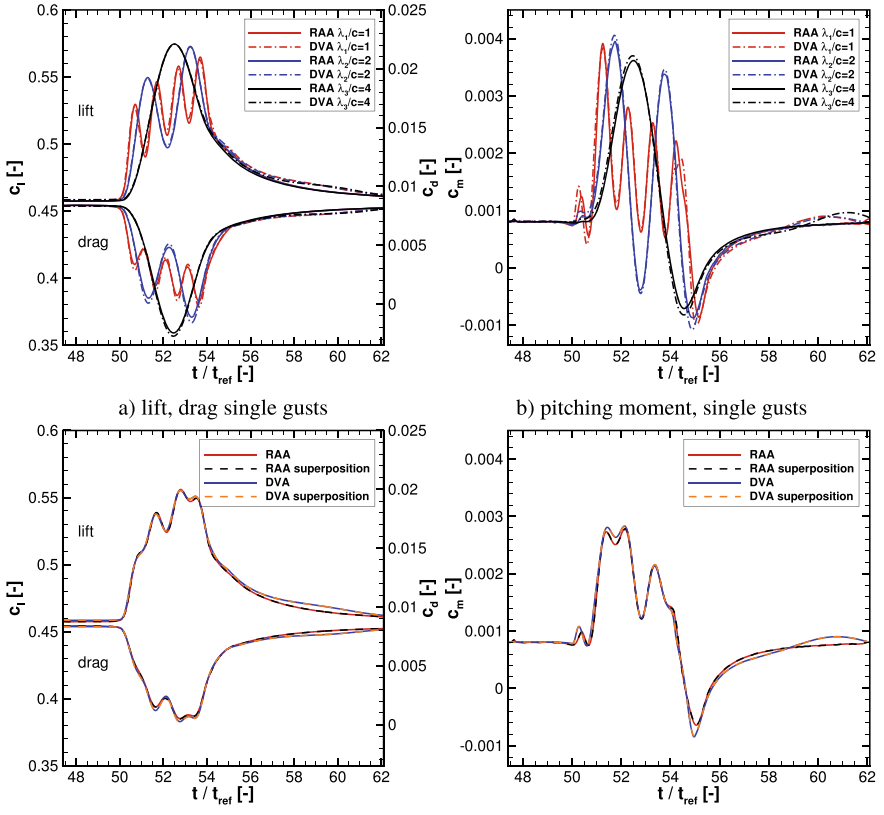
leading edge. Although Δx increases noticeably, the increase is much smaller than it is for the angle of attack variation from $\alpha = 4°$ to $\alpha = 10°$. The reduced surface pressure gradients of the NACA 0006 downstream from the suction peak result in a comparable Δx of NACA 0006 and NACA 0012 one chordlength downstream from the trailing edge. Regarding lift and drag coefficient shown in Fig. 5a, the DVA results of the NACA 0006 airfoil show a better agreement with the RAA results than for the NACA 0012 airfoil. The relative error of the maximum lift reduces from 1.18% to 0.54%. The absolute differences between DVA and RAA at the extreme points of lift and drag decrease as well as shown in Table 3. This is also the case for the maximum of the pitching moment $\Delta c_{m,max}$. The time difference of the extreme points between RAA and DVA decreases as well for the NACA 0006 airfoil. Only for $t/t_{ref} > 51.25$ after $c_{m,max}$ is reached the DVA pitching moment of the NACA 0006 shows higher deviations to the RAA than the NACA 0012. Overall, the higher surface pressure gradients downstream from the suction peak at the NACA 0012 airfoil have a more significant influence on the gust shape variation in the RAA simulation than the high but spatially narrow surface pressure gradients at the suction peak of the NACA 0006. This leads to an increased agreement between RAA and DVA for the NACA 0006 compared to the NACA 0012.

The difference between the NACA 0012 and the NACA 64-012 are smaller than for the NACA 0006. The DVA results of the NACA 64-012 show slightly higher deviations to the RAA at $c_{l,max}$ and $c_{m,max}$ with the relative error at the point of maximum lift increasing from 1.18% to 1.34%. The drag coefficient as well as the time differences of the extreme points between RAA and DVA remain almost constant compared to the NACA 0012. Thus, the variation of the surface pressure distribution and the shift in the maximum thickness position has only minor influences on the accuracy of the DVA.

In summary, the effect of airfoil aerodynamics on the gust shape, and accordingly the accuracy of the DVA, is less sensitive to high surface pressure gradients in a small region but more sensitive when moderate to high gradients exists over a longer distance. This leads to a better agreement of the DVA results with the RAA for the NACA 0006 airfoil than for the NACA 00012 or the NACA 64-012 airfoil.

3.4 *Superposition of Multiple Gusts with Different Wavelengths*

In reality, atmospheric disturbances usually do not consist of a discrete "1-cos" signal of a certain wavelength. Different wavelengths are superimposed, up to broadband atmospheric turbulence. The studies presented in [8] and [9] show that the DVA has a satisfying agreement with the RAA for an extruded airfoil and a swept wing in broadband atmospheric turbulence involving a broad frequency range. The scope of this section is to analyze the effect of the superposition of individual discrete "1-cos" gusts on the accuracy of the DVA compared to the more accurate RAA. Three different gusts are created, all based on the "1-cos" gust formulation described in Eq. (2) with $0 \leq x \leq 4c$. Hence, the spatial extend of all three gusts is $4c$. The first gust



a) lift, drag single gusts

b) pitching moment, single gusts

c) lift, drag superposition

d) pitching moment, superposition

Fig. 6 Time history of lift, drag, and pitching moment of a NACA 0012 airfoil at $\alpha = 4°$ encountering gusts with a length of $4c$. The interaction with single "1-cos" gusts of different wavelengths (top) and with a gust signal that is composed of the different single gusts as well as a superposition of the aerodynamic coefficients of the single gusts (bottom) is shown.

Table 4 Comparison of RAA and DVA results for the composed gust signal consisting of several superimposed wavelengths. Comparison of the interaction with the composed gust signal to a superposition of the aerodynamic coefficients of the interaction with the separated gust signals (sup.).

Profile	Δc_l	err_{c_l}	Δc_d	$\Delta c_{m,max}$	$\frac{\Delta t_{c_l}}{t_{ref}}$	$\frac{\Delta t_{c_d}}{t_{ref}}$	$\frac{\Delta t_{c_{m,max}}}{t_{ref}}$
RAA vs DVA	0.00071	0.13%	0.000259	−0.000052	0.010	0.005	0.010
RAA vs RAA sup.	−0.00005	0.01%	0.000039	−0.000005	0.0	0.015	−0.030
DVA vs DVA sup.	−0.00003	0.01%	0.000028	0.000006	0.0	0.010	−0.005

consists of four consecutive "1-cos" gusts of wavelength c. Therefore, w_1 in Eq. (2) is calculated with $\lambda_1 = c$. The second gust consist of two consecutive "1-cos" gusts with $\lambda = 2c$ and the third gusts is a single "1-cos" gust with $\lambda = 4c$. The amplitude of all gusts is set to $A_{gust} = \frac{1}{30} u_\infty$. The separated effects of the three different gusts on the accuracy of the DVA compared to the RAA are evaluated for the NACA 0012 airfoil at $\alpha = 4°$ with lift, drag, and pitching moment presented in Fig. 6a and 6b. Subsequently, the interaction of the NACA 0012 airfoil with a composed gust, where the gust velocity w corresponds to a superposition of the three individual gusts with $w = \frac{1}{3} w_1 + \frac{1}{3} w_2 + \frac{1}{3} w_3$ is analyzed using RAA and DVA. The RAA and DVA result of this interaction is compared to a superposition of the aerodynamic coefficients c_l, c_d, and c_m of the three separated gusts with $c_{i,sup.} = \frac{1}{3} c_{i,1} + \frac{1}{3} c_{i,2} + \frac{1}{3} c_{i,3}$. The results of the superimposed gust and the superimposed aerodynamic coefficients for RAA and DVA are presented in Fig. 6c and 6d as well as Table 4. The curves of the superimposed coefficients are indicated by the description *sup.* or *superposition*.

The deviations in the RAA and DVA time histories of the aerodynamic coefficients of the three separated gusts in Fig. 6a and 6b are comparable to the results in Sect. 3.1. With increasing λ the DVA results agree better with the RAA results. It should be emphasized that the deviations between RAA and DVA pitching moment of the four consecutive "1-cos" gusts with $\lambda_1 = c$ are comparable to the single "1-cos" gust with $\lambda = c$ presented in Sect. 3.1 when the front of the gust interacts with the leading edge and when the gust leaves the airfoil. In between the agreement of RAA and DVA increases. This shows that the airfoil's aerodynamic changes the shape at the endpoints of a continuous gust in a similar way than a discrete signal. In between the shape of a continuous signal is less influenced by the airfoil's aerodynamic.

The results of the composed gust signal with $w = \frac{1}{3} w_1 + \frac{1}{3} w_2 + \frac{1}{3} w_3$ presented in Fig. 6c and 6d show a good agreement of RAA and DVA simulation especially for c_l and c_d. As listed in Table 4 the relative error for $c_{l,max}$ between RAA and DVA is only 0.13%. This is significantly smaller than the error in the previous sections. The c_m deviations of RAA and DVA are also considerably smaller than for the discrete gusts in Sects. 3.1 to 3.3. The results in Fig. 6c and 6d and Table 4 also show that it is possible to represent interaction of a gust consisting of several different discrete cosine signals with an airfoil by superimposing the aerodynamic coefficients of the

interaction of the separated discrete gust signals. Table 4 illustrates that this holds true for both RAA and DVA at the selected inflow conditions.

4 Scaling Test

The performance of a CFD code on a high performance computing system such as the HPE Hawk is of central interest regarding time reduction and the effective use of the available resources. The DLR TAU code already showed good scaling capabilities on other HPC systems such as Hazel Hen [8, 10]. Since Hawk represents a new generation of supercomputers with a completely different hardware architecture compared to previous clusters, a scaling test is even more important to assess the code's performance on this machine. The performance of the DLR TAU code on Hawk has not yet been investigated and is subject of the following section by means of a strong scaling test.

The NASA Common Research Model (CRM), a generic contemporary transport aircraft configuration is chosen as test case for the scaling behavior of the TAU code. The hybrid CFD mesh contains approx. 111 mio grid cells including a refined area of hexahedral cells in the wing wake region for resolving separated wake flows. This test case was chosen as its size represents a good reference for the size of the future grids used in the SCBOPT project. The scaling test is performed with the DLR TAU code version 2018.1.0 and comprises stationary RANS simulations with a second order central differencing scheme. The grid is partitioned with the chaco partitioner, creating 8 different grids with varying numbers of subgrids ranging from 256 partitions to 32768 partitions. The simulations range from 2 nodes and 256 cores for the former up to 256 nodes and 32768 cores for the latter number of partitions, respectively. For each grid, the walltime in the main solver loop is determined from the average of 4 similar simulations of 5000 iterations.

The resulting scaling behavior is shown in Fig. 7. The reference of an ideal scaling is provided as dashed blue line. An almost ideal scaling behavior from 1024 up to 16384 cores is clearly visible. While the slope of the graph decreases towards the 32768 case, the code still shows satisfactory scaling behavior in this area of approx. 3400 points per domain.

As pointed out by HLRS, the memory bandwidth might limit the performance of the code in use. Therefore, it is recommended to test if the use of not all available cores within each node is beneficial for runtime reduction. The omplace tool enables to select a specific sequence of cores within a node and to equally distribute these cores over the available resources. With respect to Hawk's topology the different settings, listed in Fig. 8 were chosen for the present investigation. The baseline tests 6,7, and 8 are performed on all available 128 cores per node. The 7omp and 8omp tests are conducted on 128 nodes and 256 nodes respectively while holding the number of processes constant at 8192. This is done with the omplace tool by setting the stride to 2 in the 7omp and 4 in the 8omp case. To be precise, a stride of 2 leads to a usage of every second core, which means 2 cores per CCX. A stride of 4 leads to

Case	Nodes	Processes	Points/Domain
1	2	256	432800
2	4	512	216400
3	8	1024	108200
4	16	2048	54100
5	32	4096	27050
6.	64	8192	13520
7	128	16384	6760
8	256	32768	3380

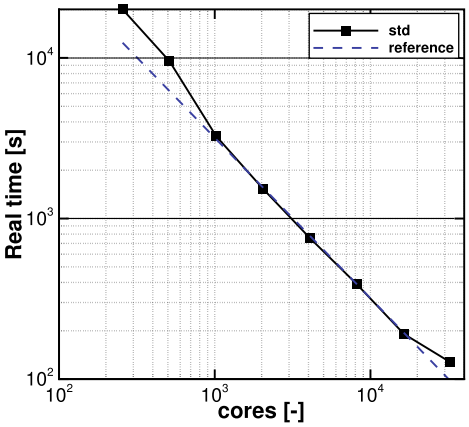


Fig. 7 Test cases and scaling behavior of the DLR TAU code on Hawk. Ideal scaling behavior plotted as dashed blue line.

Case	Nodes	Processes	Points/Domain
6	64	8192	13520
7	128	16384	6760
8	256	32768	3380
7omp	128	8192	13520
8omp	256	8192	13520

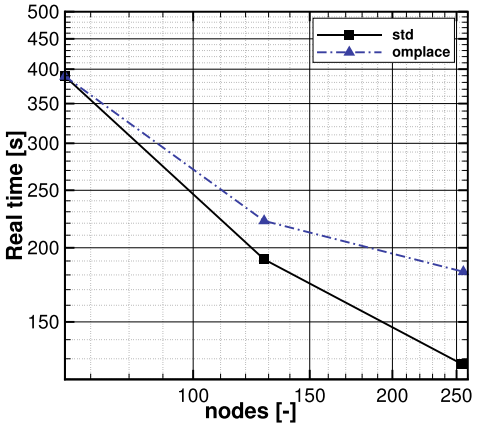


Fig. 8 Test cases and scaling behavior of the DLR TAU Code on Hawk with the omplace tool.

the selection of every forth core, which leads to one core per CCX. Figure 8 shows the wallclock time dependent from the number of nodes. The black line denotes the standard testcase with all available cores used in each node. Starting from 8192 cores and a runtime of approx. 390 s the runtime decreases with increasing number of cores to $t = 190$ s at 16384 processes and $t = 127$ s at 32768 processes. The blue dash-dotted line depicts the influence of increasing node number while keeping the number of processes constant at 8192 with omplace. The walltime also decreases significantly to approx. 220 s at 128 nodes and 180 s at 256 nodes. However, the benefit is smaller compared to the standard settings. This hints at a deceleration of the code due to limited memory bandwidth when using all available cores within one

node. Nevertheless, the benefit from higher computational power when increasing the number of nodes and cores outweighs this memory bottleneck.

This evaluation aims at an optimization regarding minimum runtime, which is desirable with regard to maximum utilization of high-performance computing resources. The results show that using more nodes up to approx. 3000 grid points per core is the most time efficient way to perform simulations with the DLR TAU code on the HPE Hawk as the code still scales well in this area. If other factors like the reduction of power consumption are taken into account, the result might be different. For example, halving the number of cores used per node only leads to an increase in wall time of 16%, while power consumption is likely to be reduced significantly.

5 Conclusion

The applicability of the simplified Disturbance Velocity Approach (DVA) regarding an airfoil encountering a discrete vertical gust was analyzed in this study. Results of CFD simulations using the more accurate Resolved Atmosphere Approach (RAA), where the gust is resolved and propagated in the flow field were compared to CFD simulations using the DVA, which does not consider the effects of the airfoil flow on the gust. All simulations were conducted for $M = 0.25$ and $Re = 11.6 \cdot 10^6$. Considering a NACA 0012 airfoil encountering a "1-cos" gust, the length scale of the gust and the angle of attack was varied. At wavelengths of $\lambda/c = 2$ and $\lambda/c = 4$ the results of the DVA agree well with the more accurate RAA in terms of lift and drag. However, small deviations between RAA and DVA exist for the pitching moment, where the changes in the gust shape due to the interaction with the airfoil have a stronger impact. These changes in the gust shape cannot be covered by the DVA. For $\lambda/c = 1$ also lift and drag of the DVA simulation show deviations to be considered. The angle of attack variation shows that the agreement of the DVA with the RAA increases with increasing angle of attack in terms of maximum lift prediction. However, the deviations of the DVA regarding drag and pitching moment increase significantly at $\alpha = 10°$ compared to $\alpha = 4°$ which limits the DVA's applicability at high angles of attack. Variation of the airfoil shape show that the accuracy of the DVA is not significantly influenced by high surface pressure gradients acting in a small region but is more prone to moderate to high gradients present over a longer distance. This decreases the accuracy of the DVA compared to the RAA results. Furthermore, it was shown that the DVA provides satisfying results for a gust signal where several wavelengths are superimposed. It is also possible to reproduce the effects of this gust signal on an airfoil by simulating the interaction with gusts of the separated wavelengths and a subsequent superposition of the aerodynamic coefficients for both gust simulation approaches. A strong scaling test showed good scaling capabilities of the DLR TAU code on the HPE Hawk. It was shown that even though the memory bandwidth limits the code's speed, using all available cores is still beneficial with respect to a reduced runtime.

Acknowledgements The authors gratefully acknowledge the Federal Ministry for Economic Affairs and Energy, which funded the work presented in this report as part of the LuFo project VitAM-Turbulence. Also, we acknowledge the High Performance Computing Center Stuttgart (HLRS) for the provision of computational resources and the continued support.

References

1. European Union Aviation Safety Agency. https://www.easa.europa.eu/sites/default/files/dfu/CS-25%20Amendment%2024.pdf, *Certification Specifications and Acceptable Means of Compliance for Large Aeroplanes CS-25, Amendment 24* (2020). Accessed 12 May 2020
2. C. Wales, C. Valente, R. Cook, A. Gaitonde, D. Jones, J.E. Cooper, The future of non-linear modelling of aeroelastic gust interaction. in *2018 Applied Aerodynamics Conference*, June 2018
3. D. Friedewald, R. Thormann, C. Kaiser, J. Nitzsche, Quasi-steady doublet-lattice correction for aerodynamic gust response prediction in attached and separated transonic flow. CEAS Aeronaut. J. **9**(1), 53–66 (2018)
4. R. Heinrich, L. Reimer, Comparison of different approaches for gust modeling in the CFD code Tau. in *International Forum on Aeroelasticity & Structural Dynamics*, June 2013
5. C. Wales, D. Jones, A. Gaitonde, Prescribed velocity method for simulation of aerofoil gust responses. J. Aircr. **52**(1), 64–76 (2015)
6. D. Schwamborn, T. Gerhold, R. Heinrich, The DLR TAU-code, recent applications in research and industry. in *European Conference on Computational Fluid Dynamics ECCOMAS CFD*, September 2006
7. R. Heinrich, L. Reimer, Comparison of different approaches for modeling of atmospheric effects like gusts and wake-vortices in the CFD code tau. in *International Forum on Aeroelasticity & Structural Dynamics*, June 2017
8. J. Müller, M. Ehrle, T. Lutz, E. Krämer: Numerical simulation of the FNG wing section in turbulent inflow. In *High Performance Computing in Science and Engineering '19*, ed. by W.E. Nagel, D.H. Kröner, M.M. Resch (Springer, Cham, 2021), pp. 435–450
9. J. Müller, T. Lutz, E. Krämer, Numerical simulation of the swept FNG wing in atmospheric turbulence. in *AIAA Aviation 2020 Forum*, June 2020
10. K. Wawrzinek, T. Lutz, E. Krämer, Numerical simulations of artificial disturbance influence on a high lift airfoil. in *High Performance Computing in Science and Engineering '17*, ed. by W.E. Nagel, D.H. Kröner, M.M. Resch (Springer International Publishing, Cham, 2018), pp. 323–337

3D Predictions of the Primary Breakup of Fuel in Spray Nozzles for Aero Engines

T. F. Dauch, G. Chaussonnet, M. C. Keller, M. Okraschevski, C. Ates, R. Koch, and H.-J. Bauer

Abstract Primary breakup of liquid fuel in a realistic fuel spray nozzle as utilized in aero engines is numerically investigated. As grid based methods exhibit a variety of disadvantages with regard to multiphase flows, the "Smoothed Particle Hydrodynamics" (SPH)-method is employed. The suitability of the method to analyze breakup of fuel has been demonstrated in recent publications. In the current contribution the methodology enabling the computation of fuel atomization in the close vicinity of the nozzle is briefely introduced. Special emphasis is put the aspects of High Performance Computing. The domain decompositioning is analyzed and scaling tests are performed. Finally, selected results are presented, which for the first time in the world give insights into the film flow on the prefilmer, the associated wetting effects and the time dependent formation of ligaments downstream the atomizing edge.

1 Introduction

Academia and industry both invest in research aiming at a reduction of pollutant emissions of air traffic. As pointed out by Lefebvre [1], Jones et al. [2] and Knudsen et al. [3] the quality and placement of the fuel spray injected into aero engines is crucial in this context. Due to challenging aerothermal and geometric conditions in the close vicinity of the fuel spray nozzle, detailed measurements at actual engine conditions are impossible. Even experiments under scaled conditions and in simplified geometries such as the one presented by Meier et al. [4] are not suitable for a detailed analysis of the fuel spray. Hence, numerical investigations of the local two-phase flow are desired giving insights into the process of primary breakup and its sensitivity to different operating conditions.

For this purpose a numerical code based on the Lagrangian "Smoothed Particle Hydrodynamics" (SPH)-method has been developed at the "Institut für Thermische

T. F. Dauch (✉) · G. Chaussonnet · M. C. Keller · M. Okraschevski · C. Ates · R. Koch · H.-J. Bauer
Institut für Thermische Strömungsmaschinen, Karlsruher Institut für Technologie, Kaiserstr. 12, 76131 Karlsruhe, Germany
e-mail: thilo.dauch@kit.edu

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_27

Strömungsmaschinen" (ITS) by Höfler et al. [5], Braun et al. [6] and Koch et al. [7]. Conventional grid-based approaches such as the Level-Set (LS) or the Volume-of-Fluid (VoF) method suffer from various disadvantages regarding the tracking and reconstruction of the phase interface, which can be avoided by SPH. Recent publications by Braun et al. [6], Dauch et al. [8] and Keller et al. [9] demonstrate the suitability of the method for predicting multiphase flows in technically relevant applications.

Previous work demonstrated the necessity of resolving the local two-phase flow and relevant eddies close to the phase interface in order to capture the relevant physical effects. This implies high computational effort. In the present paper, a methodology is presented aiming for the pareto optimum between domain size minimization and, preferably, preserving all relevant physical phenomena at the same time. The methodology is used in order to investigate the primary breakup inside a realistic aero engine fuel spray nozzle under realistic operating conditions. The focus of the current paper is particularly on aspects of High Performance Computing associated with the methodology. Selected results are presented, which give detailed insights into the formation of the liquid film, its atomization and the subsequent generation of ligaments at the atomizing edge under realistic conditions.

2 Numerical Method

The "Smoothed Particle Hydrodynamics"(SPH)-method originates from astrophysics as presented by Gingold and Lucy [10, 11]. Nowadays, SPH is employed for the analysis of different problems in science and engineering as pointed out for example by Chaussonnet [12], Keller et al. [13] and Shadloo et al. [14]. In contrast to conventional grid-based CFD methods, SPH does not require a computational mesh. Instead, the flow domain is discretized by means of moving points of mass called "particles".

The main advantage of SPH in comparison to grid-based methods with regard to multiphase flows is the inherent advection of the phase interface avoiding interface diffusion. As pointed out by Braun et al. [15] it also offers advantages with regard to code parallelization. Advanced post-processing methods suitable for the analysis of primary atomization such as presented by Dauch et al. [8] also profit from numerical results in the Lagrangian frame of reference.

The main challenge, the method is facing, is its inherent instability, which is called "Tensile Instability" as discussed by Swegle et al. [16]. Currently there is no general solution to this problem. Furthermore, the consistency of the SPH approximation is not inherently guaranteed as elaborated for example by Litvinov et al. [17]. Despite these known issues, various publications demonstrate the suitability of the method to resemble technically relevant flow configurations as presented by Koch et al. [7] and Shadloo et al. [14].

In the present study the fluid dynamics are described by the isothermal Navier-Stokes equations and associated boundary conditions. The material derivatives of density ρ and velocity $\vec{v}$ are defined by the continuity and momentum equation.

With t indicating time, p as pressure, τ as shear stress and $\vec{f}$ as volume force it yields:

$$\frac{D\rho}{Dt} = -\rho\left(\vec{\nabla}\cdot\vec{v}\right), \tag{1}$$

$$\rho\frac{D\vec{v}}{Dt} = -\vec{\nabla}p + \vec{\nabla}\cdot\tau + \rho\vec{f}. \tag{2}$$

In order to illustrate the formalism of the SPH-method, the formulation of any function f in terms of the convolution integral is utilized with δ as the Dirac-δ function at position $\vec{r}$:

$$f(\vec{r}) = \int f(\vec{r'})\delta(\vec{r}-\vec{r'})d\vec{r'}. \tag{3}$$

In a next step the Dirac function can be approximated by a so called kernel function $W(\vec{r}-\vec{r'})$. In a second step the integral is approximated by a discrete sum across finite discretization points inside a circle of influence. As a result a so called quadrature is obtained for any function f:

$$\langle f\rangle_a = \sum_b f(\vec{r}_b)W(\vec{r}_a-\vec{r}_b,h)V_b. \tag{4}$$

Equation 4 implies that any quantity $\langle f\rangle_a$ at position a can be computed as a weighted sum of the function $f\ (r_b)$ of all other neighboring particles b. The brackets $\langle\rangle$ indicate the SPH approximation. The quantity h is called the smoothing length. The volume of each particle is represented by V. The typical shape of a qunitic kernel function $W(\vec{r}_a-\vec{r}_b,h)$ as used in this study is depicted in Fig. 1 as a function of the non-dimensional coordinates x/h and y/h. At the bottom of Fig. 1 the radius of the circle of influence is illustrated.

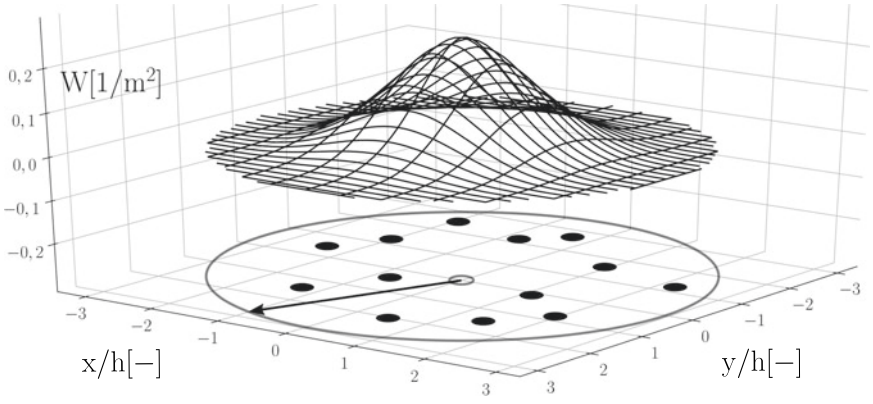


Fig. 1 Kernel function and radius of influence with neighboring particles

In a similar way, a quadrature for the gradient of f can be derived. The quadratures are used in order to approximate the terms on the right-hand-side of Eq. 1 and Eq. 2. A variety of SPH schemes exists each with different advantages and disadvantages. As the focus of the current paper is on High Performance Computing, please refer to Dauch et al. [18] for further details on the selection of physical models and SPH quadratures.

For the gradient of the pressure p a discretization scheme proposed by Monaghan [19] is incorporated. Viscosity is taken into account by means of a viscosity model based on the formulation of Cleary [20] and modified by Szewc et al. [21]. The approach by Espanol et al. [22] is used in order to compute the density field:

$$\langle \rho \rangle_a = m_a \sum_b W(\vec{r}_a - \vec{r}_b, h), \tag{5}$$

$$\left\langle -\frac{\vec{\nabla} p}{\rho} \right\rangle_a = -\frac{1}{\rho_a} \sum_b \frac{m_b}{\rho_b} (p_a + p_b) \vec{\nabla} W(\vec{r}_a - \vec{r}_b, h), \tag{6}$$

As equation of state the Tait-Volume-Equation by Cole and Weller [23] is employed. Following Wieth et al. [24], a modified version of the surface tension model by Adami et al. [25] is used. Static and dynamic contact angles are predicted based on the prescribed surface tension coefficients. According to Young's equation, static contact angles of 90° are recovered for the present case.

A variety of methods for wall boundary treatment exists, which enable the consideration of solid wall boundaries. According to Violeau and Rogers [26] it can be differentiated between three types of approaches: Using fictitious particles, repulsive functions and boundary integrals. Fundamentally, all three approaches enable to provide kernel support at the wall boundary. In the present case the fictitious particle approach is applied. Among the different variants of fictitious particles, the prescribed dummy particle method has been selected due to its advantages in case of complex geometries with high curvature and sharp edges. The approach according to Takeda et al. [27] is used. Additionally, a repulsive function resembling the Lennard-Jones potential is incorporated as presented by Wieth et al. [24] in order to avoid particle intrusion.

Activities aiming for a validation of the current implementation have been carried out based on experimental results by Braun et al. [6]. The authors demonstrate the conformity of primary breakup phenomena as captured by shadowgraphy measurements and three dimensional SPH predictions.

3 Fuel Spray Nozzle and Computational Setup

Various combustor concepts are used in aero engines aiming at a minimization of pollutant emissions. For more details about the functionality and classification of

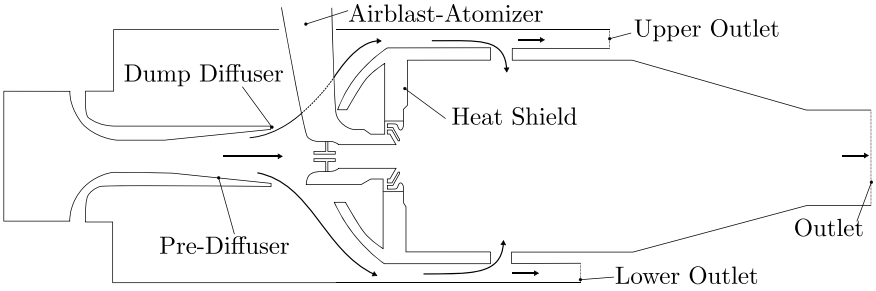


Fig. 2 Schematic of the RQL combustor

combustion concepts for aero engines please refer to Bauer [28] and Dörr [29]. Focus of the current paper is on the flow inside a "Rich-Quench-Lean" (RQL) combustor.

At the "Institute of Propulsion Technology" of the German Aerospace Center (DLR) a triple sector research combustor based on the RQL concept is operated described by Meier et al. [4]. A sketch of the rigs cross-section is displayed in Fig. 2. Air from the compressor is guided through a pre- and a dump diffuser. Further downstream a minor proportion of the air flows through the fuel spray nozzle and drives the atomization of the fuel. The design of the fuel spray nozzle investigated in this study goes back to the patent by Mansour and Benjamin [30]. The geometry of the orifice has been modified according to the patent application by Dauch et al. [31].

In the upper left corner of Fig. 3 a scheme of the nozzles cross-section is displayed. The fuel channels inside the nozzle are colored in dark gray. The arrows indicate the direction of the fuel flow. First, it enters the nozzle at the top. Further downstream it is routed into an annular duct and guided in axial direction.

The SPH domain comprises of the region around the orifice, where primary breakup takes place. Its boundaries and the cross-section of the geometry are presented in more detail on the right hand side of Fig. 3. As can be concluded from the figure, a frustoconical prefilming surface routes the fuel towards the atomizing edge. The fuel flow is surrounded by two coaxial air flows, the inner and the outer airflow. The inner airflow drives the fuel along the prefilmer.

For the numerical analysis an axisymmetric sector with a circumferential extension of 30° is considered balancing computational effort and the required circumferential size of the domain. The boundary conditions are derived from Large-Eddy Simulations of the fluid flow inside the whole combustor, which are performed prior to the SPH computations. A similar approach is presented by Sauer et al. [32], which has been transferred to the SPH context by Dauch et al. [33]. The wall boundaries are represented by at least three layers of dummy particles. Their location is obtained by incorporating the CAD2SPH approach as presented by Dauch et al. [34].

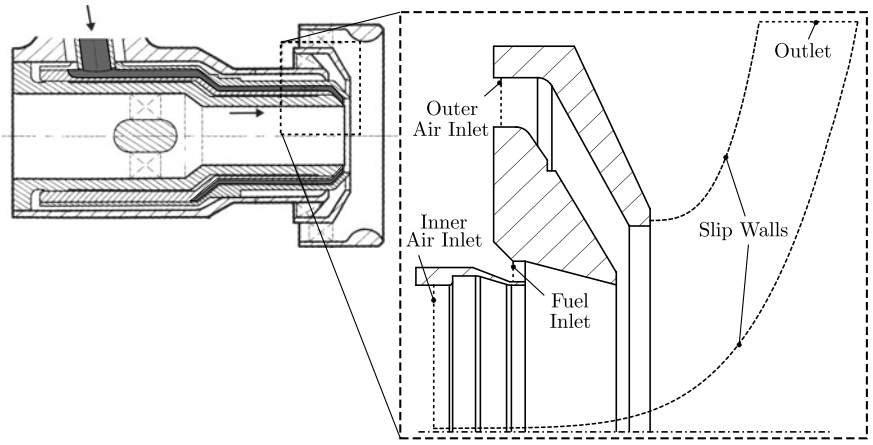


Fig. 3 SPH domain and geometry of the fuel spray nozzle according to Mansour and Benjamin [30] & Dauch et al. [31]

4 Aspects of High Performance Computing

The SPH code used for the present study is written in C++ and parallelized by means of MPI. Performance tests based on Allinea Performance Reports demonstrate clearly that the application is dominantly compute-bound.

The writing frequency is quite low and file I/O is efficiently handled by using MPI I/O. As file I/O is not critical, the utilization of many computing nodes at a time is justified aiming for a higher computational performance.

The required memory per node is dominantly determined by the number of particles per MPI subdomain. The available memory per node is sufficient for the type of SPH predictions performed in this study. However, there are two situations, in which memory becomes a limiting factor.

The first scenario occurs, if the number of compute nodes is too small related to the overall problem size. This scenario becomes relevant, if the number of nodes is kept low by intention, for example during scaling tests. The second scenario occurs in case of suboptimal domain decompositioning resulting in too many particles on some of the allocated nodes. If complex geometries are considered, domain decompositioning can become very challenging and memory overflow becomes very likely. Hence, the design of a well performing load balancing algorithm is crucial in this context.

4.1 Domain Decomposition

The parallelization concept of the SPH code is based on a pure MPI implementation. In order to provide equal load to each MPI process, the computational domain is

Table 1 Idle rates depending on the overall number of processes

10°		30°	
No. of procs.	IR	No. of procs.	IR
200	2.8	–	–
400	4.4	–	–
800	6.4	800	6.1
1200	8.1	1200	7.3
1600	9.2	1600	8.1
2000	9.6	–	–

statically decomposed during pre-processing. At the interfaces of all MPI subdomains a classical ghost particle approach is used in order to communicate information between the different MPI processes.

The decompositioning begins with the creation of a bounding box around the complete computational domain. In a next step, the bounding box is splitted into smaller rectangular cuboids. The boundaries of the cuboids are iteratively shifted in order to assign the same amount of particles to each cuboid. Each cuboid represents one MPI subdomain. Ideally, the process stops once all cuboids contain the same average number of particles per MPI process $\overline{N}_{part}$. In case of load imbalance, there is one cuboid containing the maximum number of particles per cuboid $N_{part,max}$.

A so called idle-rate IR can be defined quantifying the load imbalance:

$$IR = \frac{N_{part,max} - \overline{N}_{part}}{\overline{N}_{part}} \cdot 100\%. \tag{7}$$

In Table 1 the lowest possible values of idle rates are presented for different overall numbers of MPI processes. It is distinguished between a sector size of 10° and 30°. In both cases the domain is subdivided into two parts in z-direction. The number of splits in x- and y-direction is selected as similar as possbile. Obviously, the idle rate grows with the number of overall processes for both sector sizes. The reason for the increase is the existence of void spaces inside the bounding box. These void spaces cause increasingly more difficulties for the decompositioning algorithm to find optimal locations for the cuboids boundaries as can be concluded from the distribution of MPI cuboids illustrated in Fig. 4.

The MPI cuboids covering void spaces inside the bounding box grow very large in order to still cover a part of the domain filled with particles. The more processes are desired, the more challenging it is to place the cuboids.

On the left hand side of Fig. 4 the cuboid distribution is presented for a 2D case with only one layer of particles in circumferential direction. Contrarily, the cuboid distribution obtained for the present 3D case is depicted on the right. As can be concluded from a comparison of the cuboids in both cases, the size of the cuboids in

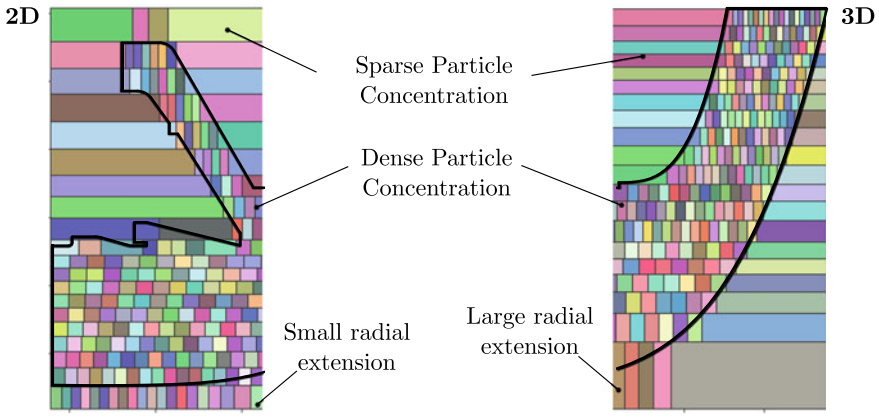


Fig. 4 Location and shape of MPI cuboids (20 × 20 subdomains in cross-section)—LEFT: 2D—RIGHT: 3D at 30°

3D decreases in radial direction. This effect can be attributed to the fact, that in 3D the number of particles in circumferential direction increases in radial direction.

The existence of void spaces inside the bounding box and the attempt to split an axisymmetric distribution of particles by means of a cartesian splitting approach both contribute to the increase of idle rates with the number of processes.

4.2 Scaling

Because the application is compute-bound, strong scaling is of primary importance for the overall efficiency. Previous scaling tests demonstrated an excellent parallel performance of the implementation as pointed out by Braun et al. [35]. Based on these findings, an additional performance increase could be achieved by further code optimization demonstrated by Braun et al. [15]. As the cross-section of the computational domain in the present paper contains void spaces inside the bounding box, it differs from the perfectly rectangular setup of Braun et al. [15]. Due to this difference, the scaling characteristics cannot be assumed to be identical.

Hence, strong scaling tests are performed in order to investigate parallel performance for the present setup. For this purpose the so called speedup S_s is evaluated. According to Eq. 8 it is defined as the relative reduction of execution time T due to an increase of the number of processes no_{procs} compared to a reference case indicated by index “ref”. Furthermore, it can be normalized in terms of the idle rate in order to take the effect of load imbalance into account as defined by Eq. 9.

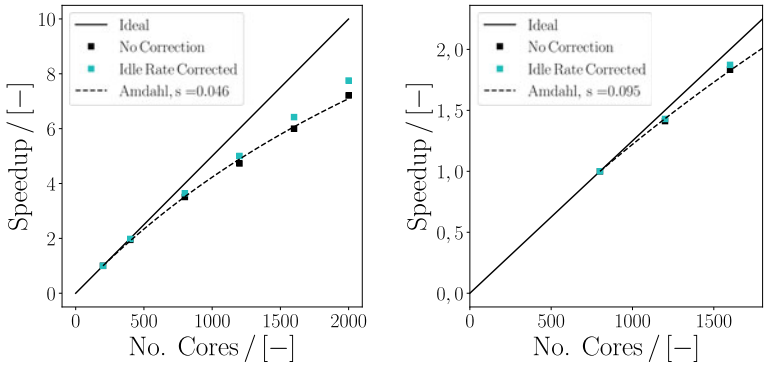


Fig. 5 Strong scaling—LEFT: 10°—RIGHT: 30°

$$S_s = \frac{T}{T_{ref}} \frac{no_{procs}}{no_{procs,ref}}, \tag{8}$$

$$S_s^* = S_s \frac{1 - IR_{ref}}{1 - IR}. \tag{9}$$

The evolution of S_s and S_s^* is investigated for the previously presented numerical setup with 10° and 30° circumferential extension. In case of the 10° sector the reference baseline is set to 200 cores and in case of 30° to 800 cores. A further reduction of the number of cores used as reference is prohibited by memory requirements. The resulting data points are illustrated in Fig. 5.

As can be concluded from the left plot in Fig. 5, the speedup S_s is close to ideal up to 800 processes. For larger numbers a performance degradation is observed following the curve defined by Amdahl's law. By fitting Amdahl's law to the data points in the scaling plot, a serial fraction of 0.046 is obtained for 10° and 0.095 for 30°. These low values of the serial fraction confirm that the setup has the potential to benefit from parallel processing according to Hager and Wellein [36]. For larger numbers of cores, the influence of the idle rate becomes more significant.

In Fig. 6 the parallel efficiency ϵ is plotted as a function of the number of MPI processes. In tune with the interpreation of the scaling plots, the parallel efficiency is higher for the 30° (beyond 70%) than for the 10° case (beyond 90%). It can be concluded from these values, that even more MPI processes could be used in case of the 30° sector without significant loss of parallel performance.

In Fig. 7 the number of particles per process is presented for each sector size. The average number of particles per process is much higher in case of the 30° sector resulting in a higher average volume to surface ratio of the MPI cuboid. As consequence, the specific communication effort is lower. The superior scaling of the 30° case compared to 10° can be attributed to this effect.

In the following, weak scaling is investigated for the 10° sector. For this study four different simulations are performed at different spatial resolutions. The total amount of particles inside the domain ranges from approximately 100×10^6 up to

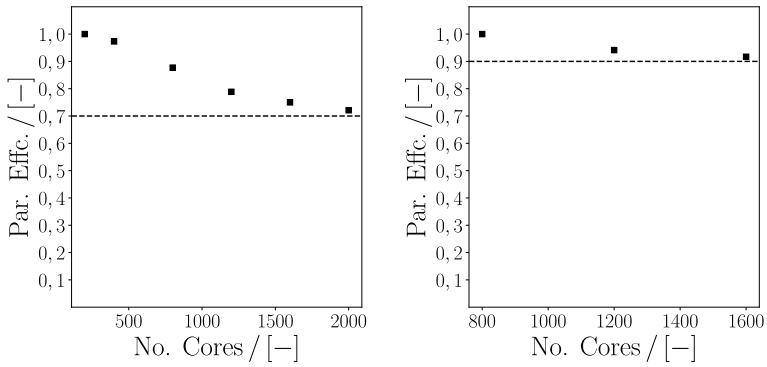


Fig. 6 Parallel efficiency—LEFT: 10°—RIGHT: 30°

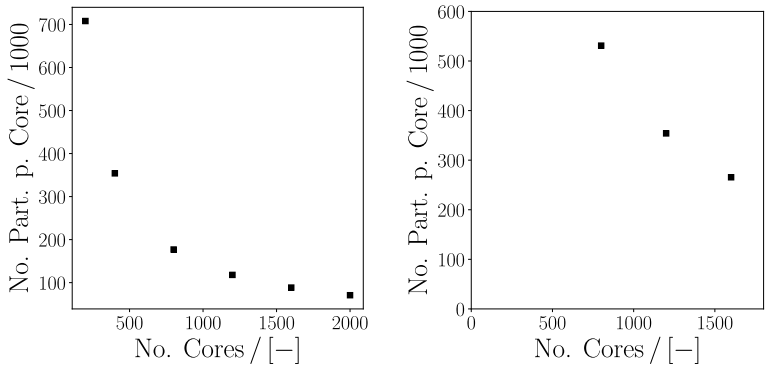


Fig. 7 Numbers of particles per process—LEFT: 10°—RIGHT: 30°

460×10^6. The amount of particles per MPI domain is kept approximately constant at around 375 000 particles per process.

In the case of weak scaling, the speedup S can be modeled by a linear function of the number of processes according to Gustafson's law. The offset of the line and its slope are determined by the serial and the parallel fraction of the runtime s and p. The serial and the parallel fraction depend on each other, because they add up to one. Hence, the speedup according to Gustafson's law can be represented as:

$$S_{Gustafson} = s + p \cdot n_{procs} = s + (1 - s) \cdot n_{procs}. \tag{10}$$

In Fig. 8 on the left the weak scaling performance is presented. According to Eq. 10 the speedup can be modeled as a line, whose slope represents the parallel fraction of the runtime. Using the datapoints as input for a linear regression yields a serial fraction of $s = 0.319$. Comparing the datapoints with and without idle rate correction in Fig. 8 on the left, confirms the little influence of load balancing on weak

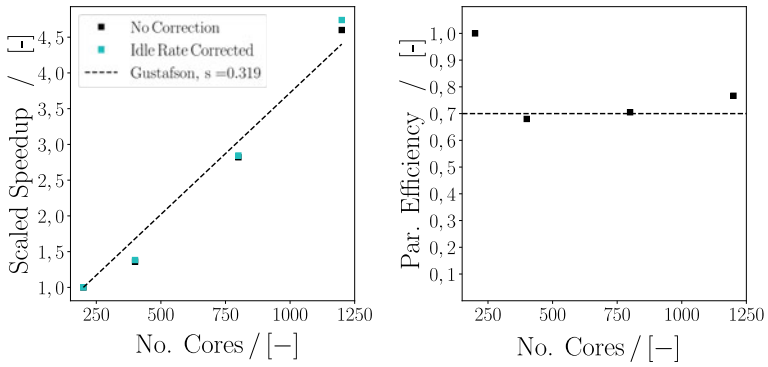


Fig. 8 10° sector—LEFT: weak scaling—RIGHT: parallel efficiency

scaling in the present case. On the right the parallel efficiency is presented. As can be concluded from the plot, it is at least close to 70% for all considered cases.

5 Results and Discussion

In the following results of the SPH predictions as obtained for the 30° sector are presented. In Fig. 9 time averaged velocity profiles at the cross-sections right upstream (Pos. A) and downstream (Pos B.) of the atomizing edge are depicted. The profile at Pos. A is relevant for the atomization process as it represents the flow velocity profile impacting the fuel film on the prefilmer. The profile at Pos. B characterizes the shear zone where the fuel is released into.

Close to the center axis at Pos. A the axial velocity is high due to the bullet as displayed in Fig. 3. Close to the wall radially outward there is another local maximum caused by the swirling motion. A high velocity gradient close to the wall at Pos. A implies that the fuel film is exposed to high shear stress emerging from the orifice. Whereas the radial velocity is approximately zero across the whole channel, the shape of the circumferential velocity profile resembles a smoothed free vortex flow.

At Pos. B the velocity close to the atomizing edge is approximately zero, because it acts as a bluff body. The swirling motion of the inner air flow drives the fluid radially outward leading to a radial shift of the axial profile from Pos. A to Pos. B. Due to the shape of the outer air shroud, the flow in the outer air duct is directed radially inward. In the vicinity of the atomizing edge the inner and the outer air flow balance each other pushing the time averaged radial velocity down to zero.

In Fig. 10 the shape of the fuel film on the prefilmer is depicted at two different points in time t_{ref} and $t_{ref} + \Delta t$. At t_{ref} the onset of an axial instabilty can be identified, which is subsequently transitioning into a growing wave crest convecting downstream as depicted on the right at time $t_{ref} + \Delta t$. As can be concluded from the

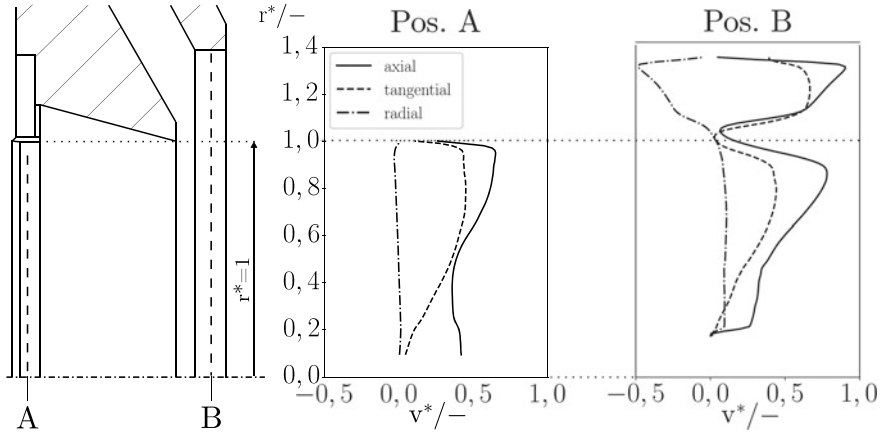


Fig. 9 Velocity profiles in the gas phase at pos. A and B

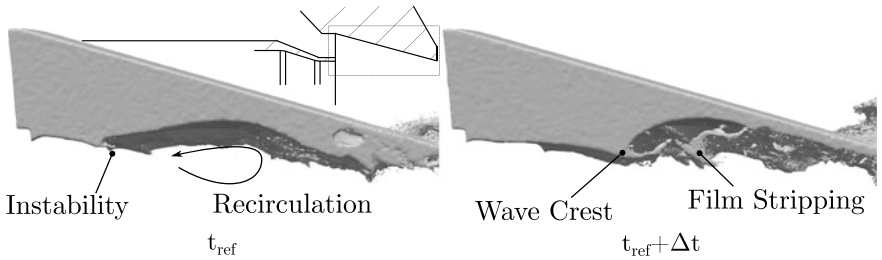


Fig. 10 Evolution of fuel film on prefilmer

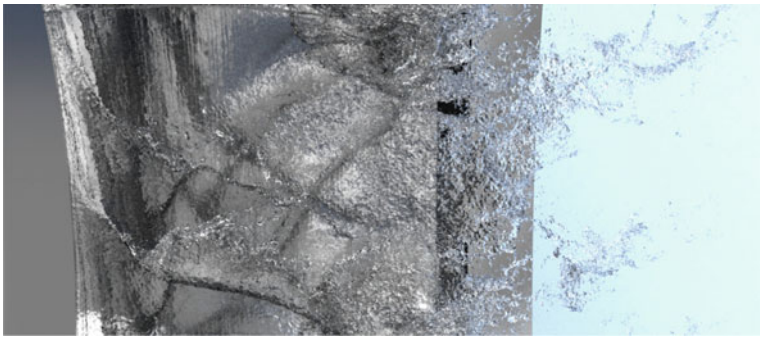


Fig. 11 Rendering of spray as predicted by SPH using alpha-shapes algorithm

illustration on the right hand side, liquid is separated from the film, which is a well known effect called"film stripping".

In Fig. 11 the spray formation as predicted by SPH is depicted for one instant of time. Three copies of the 30° sector are stitched to each other in order to improve

the visual perception. The wave crests in circumferential direction can be observed as dark lines. Downstream the atomizing edge fragments of a liquid lamella are atomized into shreds.

6 Conclusions

A methodology for the numerical prediction and analysis of primary breakup in realistic fuel spray nozzle configurations for aero engines is presented. Focus is on the domain decompositioning, the quantification of its quality and the results of extensive scaling tests for different domain sizes. The optimal number of MPI processes is derived. Finally, selected results of 3D simulations are presented as obtained from HPC predictions on ForHLR II. The main findings are:

1. Void spaces inside the bounding box around the fluid domain are a challenge to the currently implemented domain decompositioning algorithm. The development of a more generic domain decompositioning method, which is not based on rectangular MPI subdomains, is expected to improve parallel performance.
2. Strong and weak scaling tests were performed for the present numerical setup. Analyzing the serial fractions of the application according to the performance model of Amdahl and Gustafson confirms that the numerical code benefits from parallel processing.
3. Selected results of the 3D predictions give valuable insights into the local multiphase flow effects in realistic fuel spray nozzle geometries in aero engine combustors.

Acknowledgements This work was performed on the supercomputer ForHLRII funded by the Ministry of Science, Research and the Arts Baden-Württemberg and by the Federal Ministry of Education and Research. In addition the authors would like to thank Rolls-Royce Deutschland Ltd & Co KG for the outstanding cooperation.

References

1. A. Lefebvre, D.R. Ballal, *Gas Turbine Combustion*, 3rd edn. (CRC Press, Boca Raton, 2010)
2. W.P. Jones, S. Lyra, S. Navarro-Martinez, Numerical investigation of swirling kerosene spray flames using large eddy simulation. Combust. Flame **159**(4), 1539–1561 (2012)
3. E. Knudsen, Shashank, H. Pitsch, Modeling partially premixed combustion behavior in multiphase LES. Combust. Flame **162**(1), 159–180 (2015)
4. U. Meier, J. Heinze, E. Magens, M. Schroll, C. Hassa, S. Bake, T. Dörr, Optically accessible multisector combustor: application and challenges of laser techniques at realistic operating conditions, in *Proceedings of ASME Turbo Expo 2015: Turbine Technical Conference and Exposition* (2015). (GT2015-43391)
5. C. Höfler, S. Braun, R. Koch, H.-J. Bauer, Modeling spray formation in gas turbines-a new meshless approach. J. Eng. Gas Turbines Power **135**(1), 011503 (2012)

6. S. Braun, L. Wieth, S. Holz, T.F. Dauch, M.C. Keller, G. Chaussonnet, S. Gepperth, R. Koch, H.-J. Bauer, Numerical prediction of air-assisted primary atomization using smoothed particle hydrodynamics. Int. J. Multiph. Flow **114**, 303–315 (2019)
7. R. Koch, S. Braun, L. Wieth, G. Chaussonnet, T.F. Dauch, H.-J. Bauer, Prediction of primary atomization using smoothed particle hydrodynamics. Eur. J. Mech. - B/Fluids **61**(Part 2), 271–278 (2017). (Rotating Flows)
8. T.F. Dauch, C. Ates, T. Rapp, M.C. Keller, G. Chaussonnet, J. Kaden, M. Okraschevski, R. Koch, C. Dachsbacher, H.-J. Bauer, Analyzing the interaction of vortex and gas-liquid interface dynamics in fuel spray nozzles by means of Lagrangian-coherent structures (2D). Energies **12**(13), 2552 (2019)
9. M.C. Keller, S. Braun, L. Wieth, G. Chaussonnet, T.F. Dauch, R.Koch, C. Schwitzke, H.-J. Bauer, Smoothed particle hydrodynamics simulation of oil-jet gear interaction. J. Tribol. **141**(7), Art.–Nr. 071703 (2019)
10. R. Gingold, J. Monaghan, Smoothed particle hydrodynamics - theory and application to non-spherical stars. Mon. Not. R. Astron. Soc. **181**, 375–389 (1977)
11. L. Lucy, A numerical approach to the testing of the fission hypothesis. Astron. J. **82**, 1013–1024 (1977)
12. G. Chaussonnet, S. Braun, L. Wieth, R. Koch, H.-J. Bauer, Influence of particle disorder and smoothing length on SPH operator accuracy, in *10th International SPHERIC Workshop, Parma, Italy, 16th–18th June 2015* (2015)
13. M.C. Keller, S. Braun, L. Wieth, G. Chaussonnet, T.F. Dauch, R. Koch, C.Höfler, H.-J. Bauer, Numerical modeling of oil-jet lubrication for spur gears using smoothed particle hydrodynamics, in *Proceedings of the 11th International SPHERIC Workshop* (2016), pp. 69–76
14. M. Shadloo, G. Oger, D. Le Touzé, Smoothed particle hydrodynamics method for fluid flows, towards industrial applications: motivations, current state, and challenges. Comput. Fluids **136**, 11–34 (2016)
15. S. Braun, L. Wieth, T. Dauch, M. Keller, G. Chaussonnet, C.Höfler, R. Koch, H.-J. Bauer, HPC predictions of primary atomization with SPH: challenges and lessons learned, in *11th International SPHERIC Workshop* (2016)
16. J.W. Swegle, D.L. Hicks, S.W. Attaway, Smoothed particle hydrodynamics stability analysis. J. Comput. Phys. **116**(1), 123–134 (1995)
17. S. Litvinov, X.Y. Hu, N.A. Adams, Towards consistence and convergence of conservative SPH approximations. J. Comput. Phys. **301**, 394–401 (2015)
18. T.F. Dauch, T. Rapp, G. Chaussonnet, S. Braun, M.C. Keller, J. Kaden, R. Koch, C. Dachsbacher, H.-J. Bauer, Highly efficient computation of finite-time Lyapunov exponents (FTLE) on GPUs based on three-dimensional SPH datasets. Comput. Fluids **175**, 129–141 (2018)
19. J. Monaghan, Smoothed particle hydrodynamics. Ann. Rev. Astron. Astrophys. **30**(1), 543–574 (1992)
20. P.W. Cleary, Modelling confined multi-material heat and mass flows using SPH. Appl. Math. Model. **22**(12), 981–993 (1998)
21. K. Szewc, J. Pozorski, J.-P. Minier, Analysis of the incompressibility constraint in the smoothed particle hydrodynamics method. Int. J. Numer. Meth. Eng. **92**(4), 343–369 (2012)
22. P. Español, M. Revenga, Smoothed dissipative particle dynamics. Phys. Rev. E **67**, 026705 (2003)
23. R.H. Cole, R. Weller, Underwater explosions. Phys. Today **1**, 35 (1948)
24. L. Wieth, S. Braun, R. Koch, H.-J. Bauer, Modeling of liquid-wall interaction using the meshless smoothed particle hydrodynamics (SPH) method, in *26th Annual Conference on Liquid Atomization and Spray Systems* (2014)
25. S. Adami, X.Y. Hu, N.A. Adams, A new surface-tension formulation for multiphase SPH using a reproducing divergence approximation. J. Comput. Phys. **229**(13), 5011–5021 (2010)
26. D. Violeau, B.D. Rogers, Smoothed particle hydrodynamics (SPH) for free-surface flows: past, present and future. J. Hydraul. Res. **54**(1), 1–26 (2016)
27. H. Takeda, S.M. Miyama, M. Sekiya, Numerical simulation of viscous flow by smoothed particle hydrodynamics. Progr. Theoret. Phys. **92**(5), 939–960 (1994)

28. H.-J. Bauer, New low emission strategies and combustor designs for civil aeroengine applications. Progr. Comput. Fluid Dyn. **4**, 130–142 (2004)
29. T. Dörr, Combustion in aero-engines—introduction to aero-engine gas turbine combustion, in *Von Karman Institute for Fluid Dynamics—Lecture Series* (2012)
30. A. Mansour, M. Benjamin, Pure airblast nozzle. U.S. Patent and Trademark Office, Washington, September 2003
31. T. Dauch, R. Koch, H.-J. Bauer, T. Dörr, S. Bake, Nozzle for a combustion chamber of an engine (2019)
32. B. Sauer, A. Sadiki, J. Janicka, Embedded DNS concept for simulating the primary breakup of an airblasted liquid sheet. Atomization Sprays **26**(3), 187–215 (2016)
33. T.F. Dauch, S. Braun, L. Wieth, G. Chaussonnet, M.C. Keller, R. Koch, H.-J. Bauer, Computational prediction of primary breakup in fuel spray nozzles for aero-engine combustors, in *29th Annual Conference on Liquid Atomization and Spray Systems* (2017)
34. T.F. Dauch, M. Okraschevski, M.C. Keller, S. Braun, L. Wieth, G. Chaussonnet, R. Koch, H.-J. Bauer, Preprocessing workflow for the initialization of SPH predictions based on arbitrary cad models," in *Proceedings of the 12th International SPHERIC Workshop* (2017)
35. S. Braun, M. Krug, L. Wieth, C. Höfler, R. Koch, H.-J. Bauer, Simulation of primary atomization: assessment of the smoothed particle hydrodynamics (SPH) method, in *13th Triennial International Conference on Liquid Atomization and Spray Systems* (2015)
36. G. Hager, G. Wellein, *Introduction to High Performance Computing for Scientists and Engineers* (CRC Press - Taylor & Francis Group, Boca Raton, 2010)

CFD Study of an Offshore Wind Turbine in Maintenance Conditions

Marion Cormier, Patrick Letzgus, Thorsten Lutz, and Ewald Krämer

Abstract In the present study, CFD simulations of the flow around a multi-MW offshore wind turbine in maintenance position are analysed. Fully-resolved Detached Eddy Simulations of a standstill wind turbine, oriented in the L-position, are performed. Three cases are considered, covering an isolated wind turbine at the rated wind speed and isolated and non-isolated wind turbines at the cut-off wind speed. The flow physics are analysed in the hovering and approach regions of maintenance helicopters. The analysis of the turbulence kinetic energy and the flow inclination angle show no particularly high perturbations at the rated wind speed. At the high wind speed, high variations of the turbulence kinetic energy and of the inclination angle are observed for the approach manoeuvre in the wake of the nacelle.

1 Introduction

Offshore wind energy production has been growing significantly over the last decade. This success results from various aspects, like the available space, less perturbations for inhabitants and cities, and stronger and steadier winds that lead to significantly higher full-load hours offshore than onshore for a comparable system [19].

However, one of the challenges of offshore wind turbines remains their difficult access in case of failure. Due to the high energy yield of multi-MW offshore wind turbines, off-time due to maintenance operations represent higher financial losses than onshore, adding an extra economical pressure. The maintenance crew is transported to the turbine either by boat or helicopter, whereby helicopter transfer is much more time-efficient. Weather conditions strongly restrict the aerial operations and despite experience-based estimation of the flyable weather, pilots sometimes report high turbulence intensities at the turbine and need to return. The purpose of the German research project HeliOW (Hubschrauber Einsätze in Offshore Windparks—Helicopter operations in offshore wind parks) is to study the effects of wind turbine

M. Cormier (✉) · P. Letzgus · T. Lutz · E. Krämer
Faculty 6, Institute of Aerodynamics and Gas Dynamics, University of Stuttgart, Pfaffenwaldring 21, 70569 Stuttgart, Germany
e-mail: cormier@iag.uni-stuttgart.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_28

wakes on helicopter flight dynamics during maintenance operations. In this project, coordinated by the German Aerospace Center (DLR), a process chain is built up between 4 research partners. The University of Tübingen (EKUT) measures the inflow of a near-offshore wind turbine with an Unmanned Aerial Vehicle (UAV), which is then used to setup high fidelity simulations of the wake of offshore wind turbines at the University of Stuttgart. The resulting flow fields are then used in a real-time flight simulator by the DLR and in aerodynamically coupled desktop simulations by the University of Munich (TUM), in order to identify the helicopter and pilots reactions.

Wind turbine wakes have been extensively studied numerically and experimentally over the past years [20, 26]. Besides the velocity deficit caused by the energy extraction from the rotor and the local changes in wind direction, concentrated tip vortices characterise the wind turbine near-wakes. These events are seen by an approaching helicopter as abrupt changes in inflow velocity and angle that modify the flight dynamics. Further downstream the wind turbine, the tip vortex helix breaks down into small-scale turbulence, increasing the momentum of the flow field and modifying the inflow conditions of a downstream located wind turbine.

One can distinct two scenarios which can pose problems to helicopter flight dynamics during offshore maintenance operations: the perturbations which result from the wake of operating wind turbines when crossing the wind park to reach the failured wind turbine, and the hovering above the nacelle, at more than 100 m above the sea level and under strong and turbulent winds. Previous studies have addressed the first scenario [1, 3, 6, 24]. The present study focuses on the second one, the hovering above the nacelle. The particularity of this manoeuvre, in comparison to traditional hovering applications, is that the wind velocity is not slowed down within a developed boundary layer. The helicopter is exposed to high altitude winds, eventually with partial ground effects if the helicopter blades stick out the nacelle surface, and should meanwhile keep a constant position for roping crew and material. To this purpose, high-fidelity CFD flow fields of the wind turbine in maintenance configuration are generated, which will be used in a further step in helicopter flight simulators by the project partners TUM and DLR, as described above. To better understand the helicopter reactions observed in the simulators, a detailed analysis of the flow physics around the turbine is mandatory.

In the present paper, the high-fidelity flow simulations of a multi-MW generic offshore wind turbine are presented and the flow field is analysed in the hovering and approach region of the helicopter. In the following section, details of the setup and simulation cases are provided and the numerical methods used are described, including an analysis of the performance of the flow solver. Then, the regions of interest for the helicopter during hovering operations are defined and the flow physics analyzed in these regions. Finally, a short conclusion with an overview of future works is given.

2 Numerical and Computational Details

In this section, the numerical methods applied for the current study are described and details of the numerical setup are presented. Moreover, the performance of the CFD flow solver on the Hazel Hen server at the high-performance computing center in Stuttgart (HLRS) is addressed.

2.1 Numerical Methods

In the current study, the finite-volume flow solver FLOWer, originally developed at the German Aerospace Center (DLR) [12], is applied to simulate the unsteady flow field around the wind turbine. The unsteady compressible Navier-Stokes equations are solved block-wise with a dual time-stepping scheme and a sawtooth W-cycle multigrid algorithm in order to accelerate the convergence of the solution. Each turbine component is meshed separately and the structured grids are embedded within a Cartesian background mesh thanks to the Chimera technique [2].

The accuracy and capabilities of FLOWer for wind turbine simulations have been shown in several projects, among others within an international benchmark comparison of different aerodynamic codes [4], for the simulation of wind turbines in complex terrain [23], and recently for the high-fidelity simulation of wind turbine wakes with the DES method [27].

To capture the unsteady flow separation behind the square-edged nacelle (see Sect. 2.3), Delayed Detached Eddy Simulations (DDES) are performed. The Menter SST turbulence model [17] and the Smagorinsky subgrid scale model are applied. The 2^{nd} order Jameson-Schmidt-Turkel (JST) [7] and the 5^{th} order Weighted Essentially Non-Oscillatory (WENO) [22] spatial schemes are both implemented in FLOWer and can be assigned block-wise. In order to efficiently use the computational costs, the WENO scheme is applied in the area of interest, whereby the JST scheme is used in the surface boundary layers, allowing satisfying convergence in the whole computational domain.

To consider the influence of an upstream wind turbine, the Actuator Line (ACL) method is applied. Previous studies have shown that, with this method, the far wake dynamics are well reproduced at reduced computational costs, compared to an equivalent fully-resolved simulation [25]. As the present study focuses on the detailed aerodynamics of the maintenance wind turbine and only the far-wake effects of the first turbine are relevant, the ACL is considered suitable for the upstream wind turbine to optimize computational costs. The blade aerodynamic forces are estimated using the blade-element theory for each blade segment as a function of the local angle of attack, the local undisturbed wind velocity and the local lift and drag coefficients. The aerodynamic force is added in the flow field as a source term, whereby a Gaussian projection function is used for numerical stability reasons. The local angle of attack and inflow velocity are estimated from the flow field at the beginning of

each time step. As presented in [3], in order to get rid of the local blade effects on the flow field, the LineAve method from [9] has been implemented to extract the local velocity angle and amplitude. The lift and drag coefficients are taken from lookup tables, which have been computed ahead with Xfoil [5].

To mimic the atmospheric turbulence, synthetic turbulent inflow is realised as a superposition of a mean wind profile and turbulent fluctuations. The mean wind profile is imposed as a Dirichlet boundary condition at the inlet of the flow domain and the synthetic turbulent fluctuations, generated with the Mann model [16], are added as source terms downstream the inflow plane.

2.2 *Performance of the Solver*

The flow solver FLOWer is steadily further developed at the Institute of Aerodynamics and Gas Dynamics of the University of Stuttgart, not only to implement new physical models [10, 15, 27, 28], but also to improve the numerical performance. Figure 1 shows the results of two scaling tests performed on the Cray XC40 Hazel Hen. Thanks to the block-oriented structure of the solver, the whole flow field is split into structured blocks, which are equally distributed between the processors. Since [11], a node-to-node communication takes place to exchange the information between the blocks, which considerably reduces the MPI communication.

Figure 1 (left) shows the results of a strong scaling test. The total number of cells is kept constant, 134 Millions cells, and the workload is varied for each simulation case. In the reference case, the flow field is split into equal blocks of 262 144 cells. For each further case, the number of cells per blocks is halved until 8 192 cells per block are reached. With the diminution of the number of cells per block, a speed up of about 24 times can be achieved.

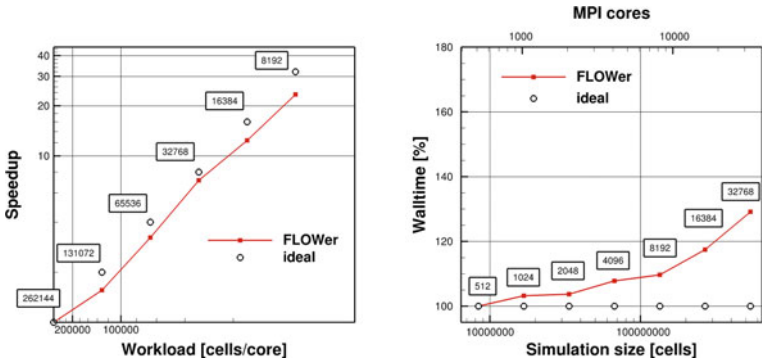


Fig. 1 Strong (left) and weak (right) scaling test of the Cray XC 40 Hazel Hen high performance computer

Table 1 Block topology of the weak scaling test

Case	Number of blocks	Number of cells per block
1	512	$16 \times 32 \times 42 = 16\,384$
2	1 024	$16 \times 32 \times 42 = 16\,384$
3	2 048	$16 \times 32 \times 42 = 16\,384$
4	4 096	$16 \times 32 \times 42 = 16\,384$
5	8 192	$16 \times 32 \times 42 = 16\,384$
6	16 384	$16 \times 32 \times 42 = 16\,384$
7	32 768	$16 \times 32 \times 42 = 16\,384$

Figure 1 (right) shows the results of a weak scaling test. On the contrary to the strong scaling test, a small initial setup is considered and its size gradually increased. The reference case contains only 512 blocks and the setup size is doubled for each further case, see Table 1. The 6^{th} case, with 16 384 blocks, is representative for the current study. In order to meaningfully compare the MPI communication between the different cases, particular attention was given to keep the block topology constant, cf. Table 1. Each block is distributed on a rank. In order to damp run time oscillations, the time required for 10 time steps with respectively 120 subiterations was averaged. As represented on Fig. 1, the performance for 16 384 cores is nearly 85%. This result is in a good agreement with tests from [14], where a good performance could be achieved in FLOWer scaling tests that were performed with more than 50 000 cores.

During the last performance workshops organized by the HLRS of University of Stuttgart, few improvements were made, which allowed to gain a few percents in the performance. The infrastructure all around Hazelhen implemented at the HLRS is well suited for the FLOWer code. Comparisons with other systems show that the best scaling is achieved on the Hazel Hen system, among others due to the use of Aries Interconnect.

2.3 Numerical Setup

A slightly modified version of the NREL 5 MW wind turbine [8, 10] is investigated in the present study. In the first part of this section, the numerical setup of the fully resolved maintenance wind turbine is presented. In the second part, the ACL setup of an upstream located wind turbine is briefly described.

2.3.1 Fully-Resolved Wind Turbine

For maintenance operations, the wind turbine is placed in the so-called “L-position”. In that position, the rotor plane is oriented parallel to the wind, whereby one of the

blades points horizontally against the wind direction. This way, the helicopter can approach against the wind, parallel to the rotor plane, and hover over the nacelle with a reduced collision risk with one of the wind turbine blades.

The NREL 5 MW wind turbine is considered, which presents a rotor radius of $R = 63$ m and a hub height of $z_{hub} = 90$ m [8]. Two reasons motivated the choice of this wind turbine: first, it is a public access reference wind turbine, second, the geometry of the nacelle, although very simple, is similar to modern square section nacelles for offshore wind turbines. Although a lot of different designs exist, this bluff body form represents a conservative case with higher perturbations expected in the nacelle wake than real modern wind turbines actually generate. The geometry of the NREL 5MW wind turbine has already been investigated with the flow solver FLOWer in the past, coupled with a structure solver [10]. The same blade, tower and blade connector grids are used in the present study, but the nacelle grid was adapted to the requirements of the current simulation case.

The block-structured grids were generated with the commercial Software Pointwise and with in-house automated scripts. For all components, the wall distance fulfills the condition $y^+ \approx 1$. All the turbine grids are embedded in a Cartesian background mesh, which features a smallest core size of $\Delta x = 1$ m around the turbine and coarsens towards the top, lateral and downstream boundaries using hanging grid nodes. Additionally, a refined Cartesian region is added in the wake of the nacelle, with isotropic cells of size $\Delta x = 0.2$ m. The setup is represented in Fig. 2, whereby

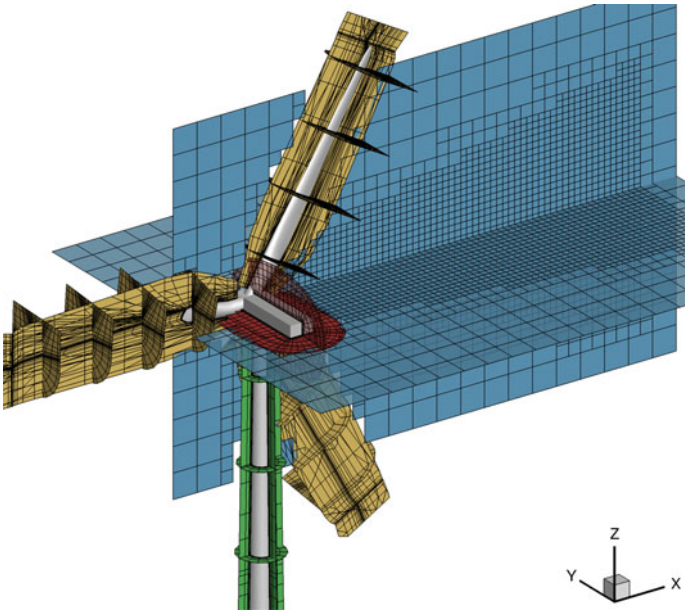


Fig. 2 Setup of the fully-resolved wind turbine in L-position. For more clarity, only each eighth mesh line is represented

the background mesh is not represented for more clarity. The computational domain extends upstream, laterally, vertically and downstream the wind turbine respectively to 360 m, 350 m, 640 m and 340 m.

2.3.2 Actuator Line Rotor

While the turbine of interest is fully resolved, the ACL approach was used to capture the far-wake effects of a neighbouring wind turbine. 99 unevenly distributed blade elements are used, with a higher density at the blade root and blade tip, where higher gradients in radial direction are expected. A comparison of the line loads for the ACL model and the fully-resolved setup of the current turbine can be found in [3].

2.4 Simulation Cases

Two wind conditions are investigated, corresponding to the wind turbine rated operating point and to the cut-off wind speed of the turbine. The first one is representative of typical operational conditions while the latter one is less likely to occur and corresponds to the maximum wind velocity at which helicopter maintenance operations are usually conducted. It representing thus an extreme case in order to identify possible critical points. In order to investigate the effects of a neighbouring wind turbine on the target nacelle to be hovered, an upstream wind turbine was modelled with the actuator line model for the high wind speed case. The inflow wind profile and turbulence characteristics were determined from large offshore databases available in the literature [18]. The main characteristics of the three different simulation setups are summed up in Table 2. The setups of the two isolated wind turbines consist of about 116 Mio. cells, while the non-isolated case required a total of 247 Mio. cells. The more than 100 Mio. extra cells in Case III are dedicated to the transport of the highly resolved wake of the upstream wind turbine to the turbine of interest.

Table 2 Overview of the simulation cases: wind velocity at hub height V_W, turbulence intensity I and length scale L used in the Mann model, rotational speed Ω of the upstream wind turbine

Case	V_W [m/s]	I [%]	L [m]	Ω [rpm]
Case I (isolated)	11.3	5	45	–
Case II (isolated)	25	6.84	60	–
Case III (non-isolated)	25	6.84	60	12.1

3 Results

In this section, the highly resolved 3D flow fields around the wind turbine are analysed. First, regions of interest for the helicopter are defined for the approach and the hovering manoeuvres. Then, the flow physics are analysed in each of these regions for all three simulated cases, with a particular focus on the perturbations that could affect the helicopter flight dynamics.

3.1 Definition of the Regions of Interest

In this section, the post-processing regions for the flow analysis related to the approach and hovering manoeuvres are defined. To determine the position and height of the helicopter rotor relatively to the wind turbine structure, macro data of an EC-135 helicopter are used. A rotor radius of $R_{heli} = 5.1$ m, a helicopter height of 3.5 m between the landing skid and the rotor and a rotational speed of $\Omega_{heli} = 395$ RPM are thus considered [13].

For the hovering case, a security distance of 4.5 m between the nacelle upper surface and the helicopter landing skid is considered. It corresponds to an altitude of the main rotor at $z_{hover} = 101$ m. The rotor center is assumed aligned with the nacelle axis and over an hypothetical landing area. The hovering region considered is represented by the circular zone in Fig. 3.

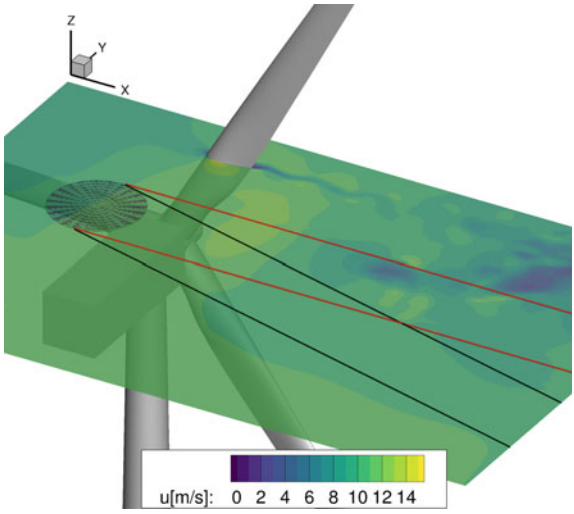


Fig. 3 Instantaneous streamwise velocity at $z = 101$ m for Case I. The black and red lines represent the approach paths A and B, respectively, and the circular zone represents the rotor during the hovering manoeuvre

For the approach scenario, the same altitude is considered as for the hovering case and two flight paths are defined. The first one, represented by the black lines in Fig. 3, forms an angle of 15° with the wind turbine rotor plane, avoiding the nacelle wake perturbations. The second one, represented by the red lines in Fig. 3, is parallel to the wind turbine rotor plane and crosses the wake of the nacelle.

3.2 Flow Physics in the Hovering Region

In this section, the flow physics in the hovering area are analysed. The data has been analysed over $t = 12$ s with a time step of $\Delta t = 0.04$ s. Considering the width of the nacelle as reference length scale for this flow, it the averaging time corresponds to 25 length scales for Case I and 50 length scales for Cases II and III. Due to uncertainty at the low frequencies in the available time window, only frequencies above 0.5 Hz are considered for the statistical analysis. For the FFT analysis, a Hann windowing function is used.

Figure 4 shows the turbulence kinetic energy (TKE) in the hovering region of the helicopter for the 3 cases. 1 to 2 orders of magnitude higher TKE values are observed in the rotor area for the cases at 25 m/s (Cases II and III) than in Case I, resulting from the higher turbulence intensity I combined to the more than twice higher inflow velocity V_W. For all cases, the TKE in the rotor plane for the hovering manoeuvre remains under 2 m^2s^{-2}. For the high wind speeds, a higher TKE can be observed in the non-isolated case, due to the far-wake turbulence generated by the upstream wind turbine.

Figure 5 shows the power spectral density (PSD) of the three velocity components for Cases II and III at the center of the helicopter rotor, giving more insights into the range of the perturbations resulting from the upstream operating wind turbine. For all velocity components, the spectrum of the non-isolated case is higher than for the isolated turbine. No peak of energy is visible at the rotational frequency of the first wind turbine 1P and its harmonic 3P, sign that the tip vortices of the first wind turbine already broke down into small-scale turbulence.

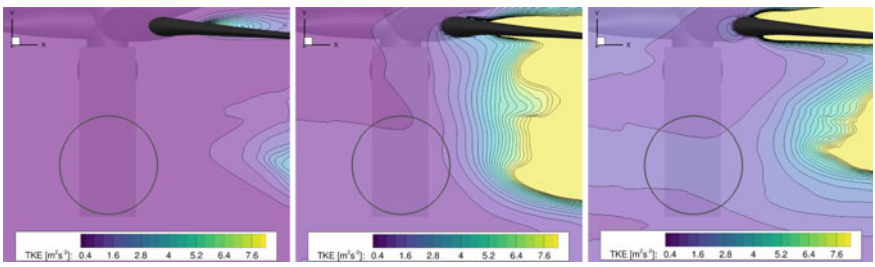


Fig. 4 Turbulence kinetic energy in the horizontal plane at height z = 101 m for Case I, Case II and Case III. The black circle represents the rotor area of the maintenance helicopter

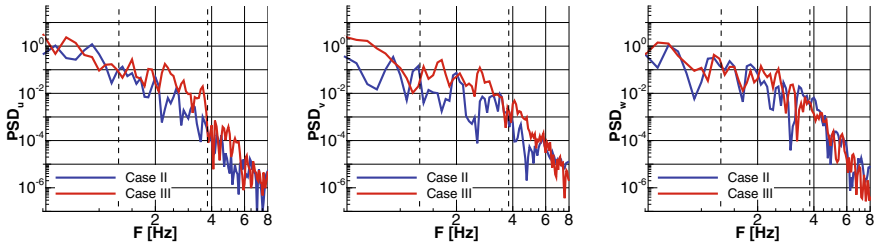


Fig. 5 Left to right: PSD of the streamwise, spanwise and vertical velocity components at the center of the helicopter rotor. The vertical dashed lines represent the 1P and 3P frequencies of the upstream rotor

3.3 Flow Physics in the Approach Region

In the L-position of the maintenance wind turbine, the wake of the square-section nacelle is characterized by the strong recirculation area and the von Kármán vortex street, as represented in Fig. 6. The flow detaches alternately from the upper and lower edges of the nacelle and oscillates upwards and downwards with the frequency f and with a downstream increasing amplitude. To characterize the oscillating flow in this region, the Strouhal number relative to the nacelle width L,$St = \frac{f \times L}{V_W}$, is evaluated for the three simulation cases. It varies little between the different scenarios, with a value of $St = 0.10$ for Case I and $St = 0.12$ for cases II and III. These results are in line with literature data [21], showing that varying the inflow velocity or slightly varying the inflow turbulence has little to no effects on the Strouhal number of square cylinders with turbulent inflow.

To analyse the amplitude of the velocity fluctuations, Fig. 7 shows the turbulence kinetic energy (TKE) for the three cases at the centerline of the previously defined paths A (solid lines) and B (dashed lines). Here again, the data has been analysed over $t = 12$ s. Consistently to the results from the previous section, at the hovering position $d = 0$ m, the highest TKE values is found in Case III. The simulated TKE augments with increasing distance d from the hovering position due to the development of the von Kármán vortex street. Moreover, the flight path through the nacelle wake (B) exhibits a TKE up to ten times higher than the lateral approach (A) for all simulation cases. It is notable that the highest energy encountered in Case I through the nacelle wake compares with the energy encountered in the lateral approach for high wind speed Cases II and III. Interestingly, higher TKEs are produced in the wake of the wind turbine for Case II than for Case III, although the opposite trend is observed in the inflow of the wind turbine of interest and above the nacelle, see Sect. 3.2, as result of the far wake of the upstream operating turbine. Reasons for the higher amplitude of the velocity fluctuations in Case II should be further analysed.

Not only the velocity fluctuations are decisive for the helicopter flight dynamics, but also the angle of attack changes seen by the blades. Figure 8 compares the flow inclination angles along A (left) and B (right) for the different scenarios. At

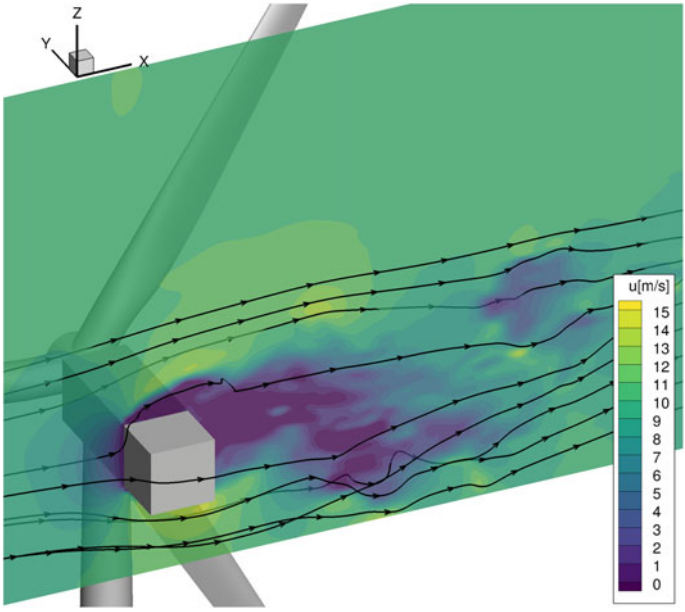


Fig. 6 Instantaneous streamwise velocity field in a vertical plane, Case I

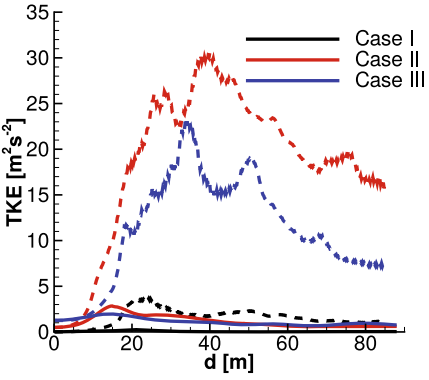


Fig. 7 Turbulence kinetic energy during the approach manoeuvre along flight path A (solid line) and B (dashed line)

the hovering point, the flow circulation around the nacelle induces a positive mean inclination angle of for all cases and maximal fluctuations of 4°. Although the mean value of the inclination angle converges with increasing distance from the nacelle to nearly 0° for all cases, the envelope of the instantaneous inclination angle α presents variations in the range $[-8° : 13°]$ and $[-30° : 80°]$ for paths A and B respectively, see Fig. 8. These high variations of the local inflow inclination angle result from the oscillating flow of the von Kármán vortex street. Such amplitudes lead to strong

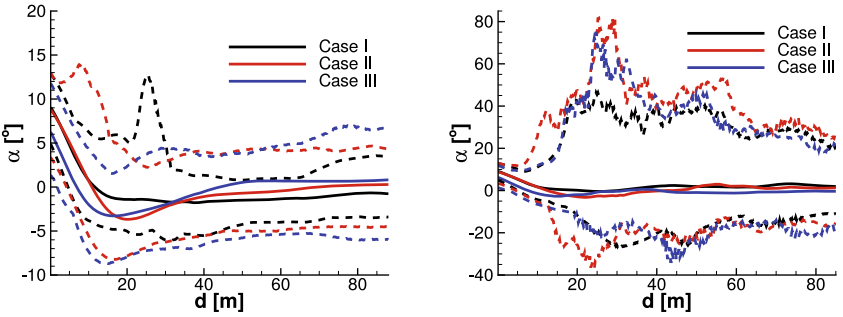


Fig. 8 Inflow angle variations during the approach flight for path A (left) and B (right).

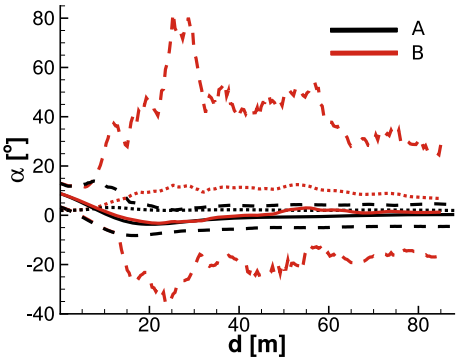


Fig. 9 Variations of the flow inclination angle during the approach flight for Case II. Solid lines: mean value, dashed lines: envelope of the instantaneous values, dotted lines: standard deviation

angle of attack changes seen by the rotating blades and, necessarily, stall is expected to occur when flying path B. Interestingly, the amplitude of the α variations of Case I observed in Fig. 8 compares with cases II and III, in contrast to the TKE observed in Fig. 7. As the local blade forces scale with the square of the magnitude of the inflow velocity, it is believed that the helicopter remains much less impacted by the flow variations in Case I than in Cases II and III.

Figure 9 highlights the influence of the flight path on flow angle for Case II, emphasizing the significant increase in the instantaneous values and in the standard deviation too.

4 Conclusion

In the present study, the unsteady flow fields around an offshore wind turbine in maintenance position were analysed. This first analysis of the flow fields allowed to

identify regions of higher perturbations for maintenance helicopters. The analysed wind turbine, the NREL 5MW, was set in the so-called "L-position", which allows the helicopter to safely hover over the nacelle platform. In that position, the side inflow of the nacelle generates an unsteady von Kármán vortex street in its wake. Three representative conditions were studied: one isolated case at the rated wind velocity $V_W = 11.3$ m/s and both one isolated and one non-isolated case at the cut-off wind speed $V_W = 25$ m/s. Based on the macro data of an EC-135 helicopter, regions of interest for the hovering and the approach manoeuvres were defined. The analysis of the flow physics in the hovering region shows an increase of the TKE with both the increase of the inflow wind speed and the consideration of an upstream located neighbouring wind turbine. No particularly high perturbations were identified in the rotor area for hovering, but high perturbations can be encountered during the approach manoeuvre. Two approach paths were considered: one parallel to the wind turbine rotor, and one with a lateral approach to the hover area. The TKE and flow inclination angles are increased of up to ten times in the parallel approach, especially at higher wind speeds where TKE can attain up to TKE $= 30\ \mathrm{m}^2/\mathrm{s}^{-2}$. However, for the $V_W = 11.3$ m/s case, the maximum magnitude of TKE in both approaches remains comparable to the values for $V_W = 25$ m/s in the undisturbed areas.

However, the interaction of the helicopter with the inflow and the real response of the helicopter to the spatial and temporal inflow variations were not taken into account. Particularly during the hovering manoeuvre, the downwash of the helicopter and partial ground effects could significantly affect the helicopter flight dynamics. Ongoing studies within the research project HeliOW with one-way and two-way coupled helicopter simulators should address this point, using the present CFD flow field as input data. A cross comparison of the reaction of both the helicopter and experienced pilots with the present flow physics study should enhance the understanding of these complex phenomena and help defining critical scenarios for maintenance helicopters.

Moreover, the performance of the flow solver on the Cray XC 40 "Hazel Hen" server of the high performance computing center in Stuttgart was addressed too, showing a good scaling on the current infrastructure.

Acknowledgements This research is funded by the German Federal Ministry for Economic Affairs and Energy (BMWi) within the framework of the German joint research project HeliOW (Grant number: 0324121). The authors gratefully acknowledge the HLRS Stuttgart for providing computational resources.

References

1. R. Bakker, A. Visingardi, B. van der Wall, S. Voutsinas, P.-M. Basset, F. Campagnolo, M. Pavel, G. Barakos, Wind turbine wakes and helicopter operations, in *44th European Rotorcraft Forum, Delft, The Netherlands* (2018)
2. J.A. Benek, J.L. Steger, F.C. Dougherty, P.G. Buning, Chimera. A grid-embedding technique. Arnold Engineering Development Center Arnold AFB TN (1986)

3. M. Bühler, P. Weihing, L. Klein, T. Lutz, E. Krämer, Actuator line method simulations for the analysis of wind turbine wakes acting on helicopters. J. Phys.: Conf. Ser. **1037–6**, 062004 (2018)
4. O. Ceyhan, K. Boorsma, A. Gonzalez, X. Munduate, O. Pires, N. Sørensen, C. Ferreira, G. Sieros, J. Madsen, S. Voutsinas, T. Lutz, G. Barakos, S. Colonia, H. Heißelmann, F. Meng, A. Croce, Latest results from the EU project AVATAR: aerodynamic modelling of 10 MW wind turbines. J. Phys.: Conf. Ser. **753** (2016)
5. M. Drela, XFOIL: An analysis and design system for low Reynolds number airfoils, in *Low Reynolds number aerodynamics* (Springer, Berlin, 1989), pp. 1–12
6. B. Horvat, M. Hajek, J. Rauleder, Analysing rotorcraft vortex encounter methods with a lattice-Boltzmann method based GPU framework, in *AIAA Scitech 2020 Forum* (2020), p. 0539
7. A. Jameson, W. Schmidt, E. Turkel, Numerical solution of the Euler equations by finite volume methods using Runge Kutta time stepping schemes, in *14th Fluid and Plasma Dynamics Conference* (1981), p. 1259
8. J. Jonkman, S. Butterfield, W. Musial, G. Scott, Definition of a 5-MW reference wind turbine for offshore system development (National Renewable Energy Lab.(NREL), Golden, 2009)
9. E. Jost, L. Klein, H. Leipprand, T. Lutz, E. Krämer, Extracting the angle of attack on rotor blades from CFD simulations. Wind Energy **21**, 807–822 (2018)
10. L. Klein, J. Gude, F. Wenz, T. Lutz, E. Krämer, Advanced computational fluid dynamics (CFD)-multi-body simulation (MBS) coupling to assess low-frequency emissions from wind turbines. Wind Energy Sci. **3**, 713–728 (2018)
11. P. Kranzinger, U. Kowarsch, M. Schuff, M. Kessler, E. Krämer, Advances in parallelization and high-fidelity simulation of helicopter phenomena, in *High Performance Computing in Science and Engineering '15* (Springer, Cham, 2016), pp. 479–494
12. N. Kroll, J.K. Fassbender, *MEGAFLOW—Numerical Flow Simulation for Aircraft Design* (Springer, Heidelberg, 2005)
13. G.J. Leishman, *Principles of Helicopter Aerodynamics* (Cambridge University Press, Cambridge, 2006)
14. J. Letzgus, L. Dürrwächter, U. Schäferlein, M. Kessler, E. Krämer, Optimization and HPC-applications of the flow solver FLOWer, in *High Performance Computing in Science and Engineering '17* (Springer, Cham, 2018), pp. 305–322
15. P. Letzgus, T. Lutz, E. Krämer, Detached Eddy simulations of the local atmospheric flow field within a forested wind energy test site located in complex terrain. J. Phys.: Conf. Ser. **1037** (2018)
16. J. Mann, The spatial structure of neutral atmospheric surface-layer turbulence. J. Fluid Mech. **273**, 141–168 (1994)
17. F.R. Menter, Two-equation eddy-viscosity turbulence models for engineering applications. AIAA J. **32**, 1598–1605 (1994)
18. A. Peña, S.-E. Gryning, J. Mann, On the length-scale of the wind profile. Q. J. Roy. Meteorol. Soc. **136**, 2119–2131 (2010)
19. S.C. Pryor, R.J. Barthelmie, Comparison of potential power production at on-and offshore sites. Wind Energy: Int. J. Progr. Appl. Wind Power Convers. Technol. **4**, 173–181 (2001)
20. B. Sanderse, S.P. Van der Pijl, B. Koren, Review of computational fluid dynamics for wind turbine wake aerodynamics. Wind Energy **14**, 799–819 (2011)
21. M. Sariouglu, T. Yavuz, Vortex shedding from circular and rectangular cylinders placed horizontally in a turbulent flow. Turk. J. Eng. Environ. Sci. **24**, 217–228 (2000)
22. U. Schäferlein, M. Keßler, E. Krämer, High order CFD-simulation of the rotor-fuselage interaction, in *44th European Rotorcraft Forum, Moscow, Russia* (2013)
23. C. Schulz, Numerische Untersuchung des Verhaltens von Windenergieanlagen in komplexem Gelände unter turbulenter atmosphärischer Zuströmung (Shaker Verlag, 2018)
24. A. Strbac, T. Martini, D. Greiwe, F. Hoffmann, M. Jones, Analysis of rotorcraft wind turbine wake encounters using piloted simulation, in *45th European Rotorcraft Forum, Warsaw, Poland* (2019)
25. N. Troldborg, Actuator line modeling of wind turbine wakes (2009)

26. L.J. Vermeer, J.N. Sørensen, A. Crespo, Review of computational fluid dynamics for wind turbine wake aerodynamics. Wind Energy **14**, 799–819 (2003)
27. P. Weihing, J. Letzgus, G. Bangga, T. Lutz, E. Krämer, Hybrid RANS/LES capabilities of the flow solver FLOWer—application to flow around wind turbines, in *Symposium on Hybrid RANS-LES Methods* (Springer, Cham 2016), pp. 369–380
28. P. Weihing, J. Letzgus, T. Lutz, E. Krämer, Development of alternative shielding functions for detached-eddy simulations, in *Progress in Hybrid RANS-LES Methods* (Springer, Cham, 2020), pp. 109–118

Transport and Climate

Markus Uhlmann

The simulations in the category "Transport and Climate" have consumed a total of approximately 110 million core-hours over the past granting period. Six projects have utilized the system HazelHen (HLRS), and eleven were dispatched on the system ForHLR II (SCC). The lions share (73%) of computational resources in this category was spent on HazelHen.

The authors of the first study ("HRCM" by Schädler et al.) report on regional climate studies with the climate version of the DWD community model COSMO-CLM. The goal is to estimate the impact of various greenhouse gas emission scenarios upon the climate of vulnerable regions in Africa in the framework of the CORDEX project. It is shown that locally very strong temperature increases as well as enhanced dry spells can be expected for most of the African continent. Further very high resolution studies with grids down to a 3km scale have been performed by the authors for the Mediterranean as well as the upper Rhine valley, allowing convection events to be directly captured, while still covering temporal intervals of a century. The outcome from the CORDEX project constitutes a key input to the IPCC assessment reports.

The authors of the second study (Warrach-Sagi et al.) make use of the Weather and Research Forecasting (WRF) model, likewise contributing to the CORDEX project run by the World Climate Research Program. Here the focus is on climate projections for Germany for a period until the end of the 21st century, employing spatial resolutions down to a 3 km grid size. The authors underline the importance of improving land-use models in order to further improve the predictive capabilities of regional climate simulations.

M. Uhlmann
Institute for Hydromechanics, Karlsruhe Institute of Technology, Kaiserstr. 12,
76131 Karlsruhe
e-mail: markus.uhlmann@kit.edu

The third project by Bauer et al. (“WRFSCALE”) aims at land-atmosphere coupling processes with the aid of large-eddy turbulence modeling in the context of the WRF platform. The authors have investigated a variety of aspects in different scenarios: from high-impact weather in the American great plains area (LAFE), land-surface modification effects in the United Arab Emirates (UAE-1), forecast of particulate matter in the Stuttgart area (OpenFCST), to assimilation of LIDAR data into their model (VAP-DA). Overall, the project “WRFSCALE” impressively demonstrates the ability of the atmospheric model system WRF to investigate atmospheric processes across scales.

In the final project in this category (“WRFHYDRO-TAG”), authored by Arnault et al., the impact of lateral terrestrial water flow upon precipitation is investigated with the aid of the hydrological version of the WRF model. The authors consider these processes at continental scales, both for Europe and Africa. In a second part, a method for tracking stable water isotopes has been implemented into the framework. Although still experimental, this capability promises to provide essential insight into land-atmospheric water pathways.

Regional Climate Simulations with COSMO-CLM: CORDEX Africa and CORDEX FPS Convection

G. Schädler, H. Feldmann, and H.-J. Panitz

Abstract The IMK-TRO at the Karlsruhe Institute of Technology (KIT) presents in the HLRS annual report for 2019 selected projects in which the CRAY XE40 "Hazel Hen" was used. Different topics have been covered in the report period. We focus in this report on our activities within the CORDEX project (CORDEX Africa and CORDEX FPS Convection) and the generation of a very high resolution regional climate ensemble. The simulations are performed with the regional climate model COSMO-CLM (CCLM) and cover spatial resolutions from 50 km to 2.8 km. The required Wall Clock Time (WCT) ranges from about 2000 to more than 300000 node-hours.

1 Overview

The working group "Regional Climate and Water Cycle" of the Institute for Meteorology and Climate Research, Department Troposphere Research (IMK-TRO) at the Karlsruhe Institute of Technology (KIT) uses the climate version of the COSMO model (COSMO-CLM, henceforth called CCLM) on the CRAY XC40 "Hazel Hen" at the HLRS high performance computing facilities to investigate past, present and future regional climate in subregions of Europe, the Arctic and Africa.

Topics of our studies include regional climate change with a focus on extremes, ensemble simulations to assess intermodel uncertainty, land surface processes, decadal predictability and very high resolution climate simulations at convection permitting scales. We are also involved in the CORDEX Flagship Projects (FPS) and CORDEX CORE activities.

G. Schädler (✉) · H. Feldmann · H.-J. Panitz
Institute for Meteorology and Climate Research, Department Troposphere Research (IMK-TRO), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
e-mail: gerd.schaedler@kit.edu

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_29

In this report, we describe in some detail two of these topics which are interesting for both their practical relevance and their numerical/computational aspects: our activities within the CORDEX Africa Project and within the CORDEX FPS Convection (CORDEX FPS-C). Section 2 describes CCLM and Sect. 3 describes the forementioned projects in more detail. Section 4 gives an overview over the resources used. The report closes with a brief outlook in Sect. 5.

2 The CCLM Model

The regional climate model (RCM) COSMO-CLM (CCLM) is the climate version of the former operational weather forecast model COSMO (Consortium for Small-scale MOdeling) of the German Weather Service (DWD). It is a three-dimensional, non-hydrostatic, fully compressible numerical model for the atmosphere and includes a land surface model. The model solves prognostic equations for wind, pressure, air temperature, different phases of atmospheric water, soil temperature, and soil water content. Further details on the COSMO model and its applications as a regional climate model can be found in [1, 2] and on the webpages of the COSMO consortium (www.cosmo-model.org) and of the CLM Community (www.clm-community.eu).

3 The CORDEX Project

CORDEX is the acronym for the COordinated Regional Climate Downscaling EXperiment (www.cordex.org, [3]), which has been initiated by the World Climate Research Programme (WCRP) in 2009 to produce standardised regional climate and climate change projections for all terrestrial regions of our planet. CORDEX also contributes to the IPCC Assessment Reports. The major goals of CORDEX are to provide a coordinated model evaluation framework, a climate projection framework, and an interface to the users of the climate simulations for climate change impact, adaptation, and mitigation studies.

Within CORDEX, there is the CORDEX CORE regional climate model integrations project providing climate and climate change data for the CORDEX CORE regions, and there are the Flagship Pilot Studies (FPS) which have been created with the aim to provide a coordinated approach to specific research questions. Our group is contributing the CORDEX CORE Africa regional climate model integration and to two of the FPS studies: one dealing with very high resolution (convection permitting) regional climate simulations (CORDEX FPS-C, [4]) and the other with the impact of land use changes on climate (CORDEX FPS LUCAS). Below, we describe our activities within CORDEX CORE Africa and CORDEX FPS-C.

3.1 CORDEX CORE Africa Simulations

Regional climate change information is needed for vulnerability, impact and adaptation studies in all parts of the world. The spatial resolution of global climate simulations is often not detailed enough for this purpose. For some continental regions like Europe, there exists already a quite large ensemble of high resolution regional downscaling simulations using various regional climate models to downscale the results of different global climate models (GCMs). However, for other regions, there are not many such downscaled results available. Within the Sixth Climate Assessment Report of the Intergovernmental Panel on Climate Change (IPCC AR6), there will be a special chapter presenting an atlas of regional climate change information for all land regions of the earth. For this purpose, the CORDEX CORE initiative was designed (https://cordex.org/experiment-guidelines/cordex-core/).

The concept of the CORDEX CORE initiative is to systematically enlarge and improve the regional downscaling ensembles for all land regions of the Earth. Each contributing modelling group should perform a downscaling of three different CMIP5 GCMs [5] to a resolution of 25 km. Two emission scenarios covering the range of potential future developments for a number of specified CORDEX regions covering all inhabited parts of the world should be used. The first mandatory emission scenario RCP8.5 (for representative concentration pathway, [6]) gives an additional radiative forcing by anthropogenic greenhouse gas emissions of 8.5 W/m^2 by 2100. It is a worst-case (although in view of the present climate state it should be called realistic, if not optimistic) scenario with no implementation of emission reduction measures, sometimes called "business-as-usual". The second emission scenario RCP2.6 is a lower emission scenario with a peak forcing of about 3 W/m^2 followed by a decline to 2.6 W/m^2 towards the end of the 21^{st} century. The three GCMs are chosen in a way that they represent low, medium and high climate sensitivity, which means a low, medium or high increase of the global mean temperature for a given increase of the greenhouse gas emissions, relative to the spread of all GCMs contributing to CMIP5.

Ensemble Design

The IMK contributes to these efforts by conducting simulations for the CORDEX Africa region (Fig. 1) with CCLM. The model version, setup and resolution for these new simulations have been changed from previous CORDEX Africa simulations (e.g. [7]). The new simulations use an updated model version with increased horizontal (0.22°, named AFR-22 instead of 0.44° for AFR-44) and vertical resolution (57 layers instead of 35). In addition, several tuning parameters have been optimized, following [8], to reduce the model bias. Test simulations indicate indeed a reduced model bias of the new simulations in all seasons and different parts of Africa compared to the older experiments (not shown). However, the systematic bias, indicated for instance by spurious shifts of the position of the inner-tropical convergence zone in the forcing GCMs, does not change in the new ensemble. Overall, ten simulations were performed for the AFR-22 ensemble with the new setup:

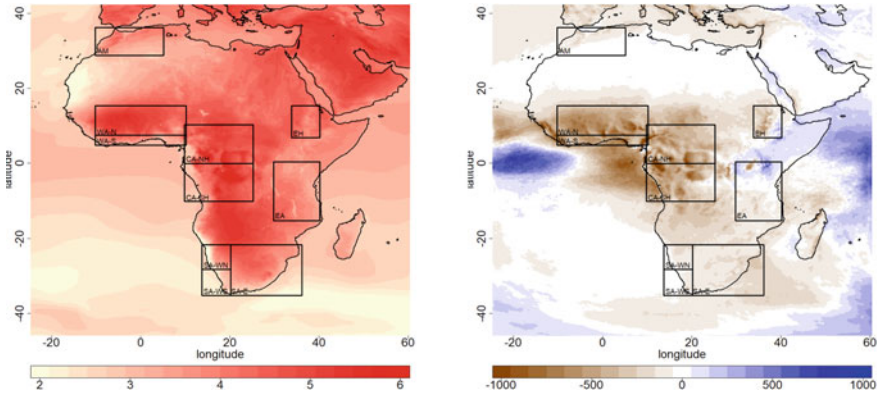


Fig. 1 End of the century climate change signal of the ensemble mean for the RCP8.5 simulations (2070–2099 minus 1971–2000). Left: Temperature change in °C. Right: Precipitation change in $mm/year$. The boxed regions indicate CORDEX Africa analysis subregions [7], notably WA-N.

- An evaluation simulation using ERA-Interim re-analysis data as boundary forcing for the period 1979–2016.
- Three "historical" simulations for the period 1950–2005 forced by the following GCMs: NorESM1-M (low climate sensitivity), MPI-ESM-LR (medium climate sensitivity) and HadGEM2-ES (high climate sensitivity).
- Three RCP2.6 scenario simulations for the period 2006–2100 using the three GCMs mentioned before.
- Three RCP8.5 scenario simulations for the same period and GCMs.

In total about 500 years have been simulated. We now discuss some results of these simulations.

End of the 21^{st} Century Climate Change Signal

Figure 1 shows the AFR-22 ensemble mean climate change signals for temperature and precipitation over Africa for the high emission scenario RCP8.5 towards the end of the 21^{st} century compared to the reference period 1971–2000. The results indicate strong changes for the western part of Africa from the Guinea coast and Sahel region (called here WA-N) over Western Central Africa to Namibia and South Africa. In these regions, the simulations indicate an increase of more than +5 °C compared to the reference period. In addition, the results show a strong reduction of the annual precipitation amounts for these regions. Dryer conditions are expected for most of Africa, except for the Horn of Africa in the Equatorial East. The ensemble spread is low to moderate in the areas of the strongest changes (not shown).

Temporal Evolution and Impact of the Scenario

Figure 2 displays the climate change signals for the individual simulations for both emission scenarios and the analysis region West Africa-North (WA-N; cf. Fig. 1). It

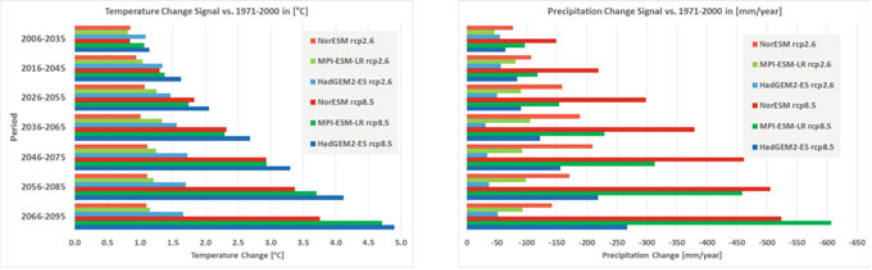


Fig. 2 Climate change signals for the WA-N region (cf. Fig. 1) for temperature (left, in °C) and precipitation (right, in $mm/year$) for 30-year periods throughout the 21^{st} century compared to the mean of the period 1971–2000.

shows the changes for 30-year climatological time periods throughout the 21^{st} century in 10 year intervals compared to 1971 to 2000. The differences between the scenarios are low to moderate in the first half and increase considerably towards the end of the projection period. For the RCP8.5 scenario, the temperature changes increase over time to about $+3.8\,°C$ resp. $+4.9\,°C$. By contrast, the changes remain between $+0.8\,°C$ and $+1.7\,°C$ for the RCP2.6 scenario. The peak changes are expected for the middle of the century with even a slight decrease towards 2100. The temperature increase over Africa is usually higher for the downscaling of the GCM with the higher global climate sensitivity (HadGEM2-ES) than for the medium (MPI-ESM-LR) and low sensitivity (NorESM1-M) ones. For precipitation, all simulations indicate a reduction over the WA-N region for all periods. As could be expected, the drying is less severe in the RCP2.6 scenario simulations than in the high emission scenario. For RCP8.5 at least two of the three GCM downscaling simulations show a strong decrease of the annual precipitation amounts.

Overall, it can be concluded that a strong reduction of greenhouse gas emissions, as in the RCP2.6 scenario, would reduce the impact of climate change. By contrast, a strong further increase of the emissions as in RCP8.5 is expected to cause severe impacts for the population and agriculture in Africa by a considerable reduction of the rain water supply and increased heat stress.

The simulations presented here will be part of the IPCC AR6 atlas of regional climate change. It will be freely available for impact, adaptation and mitigation studies.

3.2 High Resolution Simulations at Convection Permitting Scales

3.2.1 CORDEX FPS-C

The CORDEX FPS-C aims to perform first-of-its-kind ensemble climate experiments with convection permitting models to investigate present and future convective pro-

cesses and related extremes over Europe and the Mediterranean. More details and first results were presented previously in [4] and in [9].

The IMK contributed to CORDEX FPS-C by performing numerous test and sensitivity simulations. The first steps of these simulations have been described in [9]. In 2019, we began to perform the computationally expensive final high resolution simulations (resolution 0.0275°, i.e. about 3 km), aimed to study present and future convective processes and related extremes over Europe and the Mediterranean. The final high resolution simulations consist of an ERA-Interim [10] driven evaluation run for the period from 1999 to 2014 and MPI-ESM-LR driven simulations for three 10-years time slices: the present (1996 to 2005), the near future (NF, 2041 to 2050), and the distant future (DF, 2090 to 2099). The NF and DF simulations are based on the RCP8.5 emission scenario [6]. A two-stage nesting approach was used. The first nest for the ERA-Interim driven evaluation run corresponds to the EURO-CORDEX domain EUR-22; the data for this nest were taken from a longterm ERA40/ERA-Interim driven simulation at 0.22° resolution, performed as an evaluation run during the German MIKLIP project, and covering the period from 1960 to 2016. The first nest for the MPI-ESM-LR driven time slice runs corresponds to the EURO-CORDEX domain EUR-11, and the results were taken from a transient climate simulation at 0.11° spatial resolution covering the period 1950 to 2100; these simulations had been carried out during the CORDEX experiment. These first-nest data were then further downscaled to the final high resolution second nest, henceforth called ALP3.

The mean annual cycle of the 2 m temperature over the whole ALP3 domain and of the first nest EUR-22 were compared with gridded observations from E-Obs [11], which were remapped from their original resolution of 0.25° to the high resolution of 0.0275°. The high resolution FPS run shows quite a good agreement with the observations. During the winter months the model results are slightly colder (cold bias), during the summer season slightly warmer (warm bias). The overall bias is less than ±0.5 K. The results of the first nest run (EUR-22) tend to be generally warmer than the observations. As for the mean annual cycle of the monthly sums of total precipitation, both model results, ALP3 and EUR-22, overestimate the precipitation in winter, spring and autumn (wet bias), and underestimate it in summer (dry bias). Overall, the model has a general wet bias, which is slightly higher for the high resolution FPS run in the ALP3 domain.

The results of the historical 10-years timeslice run from 1996 until 2005, driven by MPI-ESM-LR can be summarised as follows: the 2 m temperature exhibits a clear cold bias (0.5 °C to 1.55 °C) for both simulations, ALP3 and EUR-11, with the EUR-11 results being even colder than ALP3. For precipitation, there is a clear wet bias for both simulations, ALP3 and EUR-11, with EUR-11 being slightly wetter. These results need to be analysed in more detail.

We summarise the climate change signal (relative to the historical control period) derived from the results of the runs described above: for 2 m temperature, we have

- nearly homogeneous warming of about 1 °C in the near future; hardly any difference between the results runs for the first (EUR-11) and second (ALP3) nests.

- further warming in the distant future ranging between 3.1 °C and 3.7 °C depending on the regions within the ALP3 domain.

and for total precipitation, we observe

- in the near future an increase of precipitation up to 13%, depending on the regions within the ALP3 domain.
- in the distant future an increase up to 10%, except for the Mediterranean coastal regions, which will become drier.

The results described here have been presented and discussed at the CORDEX FPS-C Annual meeting that took place from 26 until 27, November 2019 at Météo-France, Toulouse, France. More comprehensive analyses of the results from all participating RCMs with focus on precipitation are currently being prepared for publication.

3.2.2 Simulations for the KLIWA Project

Within the project KLIWA (Klimaveränderung und Konsequenzen für die Wasserwirtschaft), we generated a very high resolution (0.025°, about 2.8 km) regional climate ensemble focusing on an area that comprises the catchments of major rivers of the southern part of Germany (Rhine, Moselle, Danube, Inn). The data will be used by the environmental agencies of the states of Baden-Württemberg, Bavaria and Rhineland-Palatinate for a detailed assessment of the present and future hydrological regimes of these rivers and their tributaries. The simulations are carried out following a three-nest strategy. Up to now, the data of three GCMs, namely MPI-ES-LR, EC-EARTH, and HadGEM-ES2, have been downscaled first to a resolution of about 50 km (Europe) then to a resolution of about 7 km (Germany) and finally to the high resolution of the KLIWA domain [12]. The already existing high resolution ensemble is going to be extended by the regionalisation of a fourth GCM, namely CNRM-CM5. This work started in 2019 on the Hazel Hen and is now being continued on the Hawk.

4 Technical Aspects

4.1 Parallelisation and Scalability

We used horizontal domain parallelisation with tiles consisting of 45×48 grid tiles, which required 10×9 nodes. Since this had proven efficient in previous simulations, we performed no new scalability experiments. The issues of parallelisation, scalability and efficiency have been briefly discussed in earlier Annual Reports (e.g. Annual Report 2015). We plan to repeat our performance tests on the Hawk with its new architecture.

Table 1 Summary of computing demands on CRAY XC40 Hazel Hen at HLRS.

Project	Domain size	Resolution (km)	Timestep (s)	Computing time (node-h)	# of sim. years	Storage (TB)
CORDEX Africa	414 * 428 * 57	25	150	3334113	437	765
CORDEX FPS-C	522 * 490 * 50	3	25	187200	48	144
KLIWA	322 * 328 * 49	2.8	25	76000	38	19

4.2 Resources Used

Table 1 summarizes the resources used on the CRAY XC40 "Hazel Hen" at HLRS.

5 Outlook

With the availability of the Hawk for test purposes in late 2019, we began to port CCLM and its environment to the new facility without major difficulties. The high resolution CNRM-M5 simulations run currently on the Hawk. In the next months, we intend to perform scaling experiments and start regional climate simulations using the new ICON RCM and CMIP6 GCM forcing data.

References

1. M. Baldauf, A. Seifert, J. Förstner, D. Majewski, M. Raschendorfer, T. Reinhardt, Operational convective-scale numerical weather prediction with the COSMO model: Description and sensitivities. Mon. Weather Rev. **139**(12), 3887–3905 (2011)
2. B. Rockel, A. Will, A. Hense, The regional climate model COSMO-CLM (CCLM). Meteorologische Zeitschrift **17**(4), 347–348 (2008)
3. W.J. Gutowski, F. Giorgi, B. Timbal, A. Frigon, D. Jacob, H.S. Kang, K. Raghavan, B. Lee, C. Lennard, G. Nikulin, E. O'Rourke, M. Rixen, S. Solman, T. Stephenson, F. Tangang, WCRP coordinated regional downscaling experiment (CORDEX): a diagnostic MIP for CMIP6. Geosci. Model Dev. **9**(11), 4087–4095 (2016)
4. E. Coppola, S. Sobolowski, E. Pichelli, F. Raffaele, B. Ahrens, I. Anders, N. Ban, S. Bastin, M. Belda, D. Belusic, A. Caldas-Alvarez, R. M. Cardoso, S. Davolio, A. Dobler, J. Fernandez, L. Fita, Q. Fumiere, F. Giorgi, K. Goergen, I. Güttler, T. Halenka, D. Heinzeller, Ø. Hodnebrog, D. Jacob, S. Kartsios, E. Katragkou, E. Kendon, S. Khodayar, H. Kunstmann, S. Knist, A. Lavín-Gullón, P. Lind, T. Lorenz, D. Maraun, L. Marelle, E. van Meijgaard, J. Milovac, G. Myhre, H.-J. Panitz, M. Piazza, M. Raffa, T. Raub, B. Rockel, C. Schär, K. Sieck, P. M. M. Soares, S. Somot, L. Srnec, P. Stocchi, M. H. Tölle, H. Truhetz, R. Vautard, H. de Vries, K. Warrach-Sagi, A first-of-its-kind multi-model convection permitting ensemble for investigating convective phenomena over Europe and the Mediterranean. Climate Dyn. Clim. Dyn. **55**, 3–34 (2020). https://doi.org/10.1007/s00382-018-4521-8
5. K.E. Taylor, R.J. Stouffer, G.A. Meehl, An overview of CMIP5 and the experiment design. Bull. Amer. Meteor. Soc. **93**, 485–498 (2012)

6. D.P. van Vuuren, J. Edmonds, M. Kainuma, K. Riahi, A. Thomson, K. Hibbard, G.C. Hurtt, T. Kram, V. Krey, J.-F. Lamarque, T. Masui, M. Meinshausen, N. Nakicenovic, S.J. Smith, S.K. Rose, The representative concentration pathways: an overview. Clim. Change **109**(1–2), 5–31 (2011)
7. H.-J. Panitz, A. Dosio, M. Büchner, D. Lüthi, K. Keuler, COSMO-CLM (CCLM) climate simulations over CORDEX-Africa domain: analysis of the ERA-Interim driven simulations at 0.44° and 0.22° resolution. Clim. Dyn. **42**(11), 3015–3038 (2014)
8. E. Bucchignani, L. Cattaneo, H.-J. Panitz, P. Mercogliano, Sensitivity analysis with the regional climate model COSMO-CLM over the CORDEX-MENA domain. Meteorol. Atmos. Phys. **128**(1), 73–95 (2016)
9. G. Schädler, H.-J. Panitz, B. Brecht, Regional climate simulations with COSMO-CLM: CORDEX FPS convection and heat stress in cities, in *High Performance Computing in Science and Engineering 2019* (Springer, 2019), pp. 593–606
10. D.P. Dee, S.M. Uppala, A.J. Simmons, P. Berrisford, P. Poli, S. Kobayashi, U. Andrae, M.A. Balmaseda, G. Balsamo, P. Bauer et al., The ERA-Interim reanalysis: Configuration and performance of the data assimilation system. Quart. J. R. Meteor. Soc. **137**(656), 553–597 (2011)
11. M.R. Haylock, N. Hofstra, A.M.G. Klein Tank, E.J. Klok, P.D. Jones, M. New, A European daily high-resolution gridded dataset of surface temperature and precipitation. J. Geophys. Res. **113**(D20119) (2008)
12. G. Schädler, H.-J. Panitz, E. Christner, H. Feldmann, M. Karremann, N. Laube, Regional climate simulations with COSMO-CCLM: ensembles, very high resolution and paleoclimate, in *High Performance Computing in Science and Engineering 2015* (Springer, 2016), pp. 411–429

Publication and Analyses of High-Resolution Regional Climate Simulations for Europe

Kirsten Warrach-Sagi and Thomas Schwitalla

Abstract While climate change in the 21st century is a fundamentally global phenomenon, it will unfold its effects on the regional scale. There will be impacts on the regional energy and water cycles; of particular concern are changes of the intensity and frequency of extreme events such as droughts or extreme precipitation. Further the impact of climate change on yield and yield quality and crop rotations, water regime, agricultural production systems and land is concerning. Multi-model ensembles of regional climate simulations are required to make projections of the climate in the future and to evaluate the ability of models to represent extreme events. For instance, this is realized in the World Climate Research Program (WCRP) COordinated Regional climate Downscaling EXperiment CORDEX for continental scale model domains, e.g. Europe. WRFCLIM at HLRS contributed to these multi-model ensembles with simulations based on the Weather and Research Forecasting (WRF) model. The simulations were carried out within the framework of the BMBF funded Project ReKliEs-De (Regionale Klimaprojektionen Ensemble für Deutschland) for the time period 1958 to 2100 at 12 km resolution. They represent the first WRF climate projections which have been realized on the HLRS supercomputer yet and are now listed by the German Weather Service as members of the reference ensemble regional climate projections and they passed the audit of regional climate projections by the Bavarian Environmental Agency (LfU) within the KLIWA Cooperation (https://www.kliwa.de/). The spread of the ReKliEs-De ensemble results supports the confidence not only in the estimation of the evolution of medians of atmospheric variables due to climate change but also in the estimation of extreme values. The climate indices underline the importance for society and economy to mitigate climate change. In the reporting period the results were processed and published on the Earth

K. Warrach-Sagi (✉) · T. Schwitalla
University of Hohenheim, Institute of Physics and Meteorology, Garbenstrasse 30, 70599 Stuttgart, Germany
e-mail: kirsten.warrach-sagi@uni-hohenheim.de

T. Schwitalla
e-mail: thomas.schwitalla@uni-hohenheim.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_30

System Grid Federation (ESGF) ready to use for climate and climate impact studies for Europe.

1 Introduction

During the 21^{st} century, not only the global mean surface temperature but also the amount and strength of weather extremes will increase. This is mainly due to the continuous emission of greenhouse gases by human activities. Since warming over many land areas is larger than over the oceans, climate conditions will be regionally highly variable. In order to make regional projections of the climate in the future and to evaluate the ability of models to represent such phenomena, multi-model ensembles of regional climate simulations are required. This is e.g. realized in the World Climate Research Program initiated COordinated Regional climate Downscaling EXperiment CORDEX.

The objective of WRFCLIM at HLRS is to contribute to these multi-model ensembles with simulations based on the Weather and Research Forecasting (WRF) model Skamarock *et al.* [6]. Five simulations were carried out within the framework of the BMBF funded Project ReKliEs-De (Regionale Klimasimulationen Ensemble für Deutschland) for the simulation time period 1958 to 2100 at 12 km resolution. These simulations represent the first WRF climate projections which have been realized on the HLRS supercomputer yet. The results demonstrate that the ensemble members provided by the WRF provide an excellent contribution to the spread of the ReKliEs-De ensemble results. This gives a much better confidence not only in the estimation of the evolution of medians of atmospheric variables due to climate change but also in the estimation of extreme values such as hot and ice days. The corresponding climate indices provided by ReKliEs-De underline the importance for society and economy to mitigate climate change.

Furthermore, within the CORDEX Flagship Pilot Study framework simulations were carried out between 1999 and 2010 at 15 km resolution further downscaling to the convection permitting (CP) grid on 3 km in central Europe to assess mainly diurnal cycles and high impact weather. The results and the performance analyses on the CP scale will be fundamental for the set up and improvement of the next generation climate simulations. Therefore, HLRS and the University of Hohenheim are prepared to contribute to a series of national and international projects concerning seasonal and climate simulations. The results highlight the necessity of CP simulations for next generation earth system models, and multi-model ensembles to assess climate change, as the latter provide indispensable uncertainty measures for climate change signals and extremes for Germany. Thirdly the crop model GECROS was implemented into WRF-NOAHMP version 3.7.1 and applied from July 2004 to November 2005 to simulate the growing season 2005. This so-called ALCM (Atmosphere- Land surface - Crop Model) is compared to a simulation with WRF-NOAHMP, applying default table values for the leaf area index (LAI). In ALCM all grid cells with winter crops were simulated with GECROS, all other cropland

grid cells were simulated with NOAHMP's standard parameterization for croplands. The simulated summer temperatures in Germany show an improvement of temperatures by 1 °C by the dynamic crop simulation in GECROS namely in the lowlands in northern Germany. All simulations were completed at the end of 2018 but due to the large amount of data post-processing was still ongoing and required to be done on hazelhen at HLRS, since the steps required access to the data on HPSS. At the end of the reporting period the results were processed and made available for publication on the Earth System Grid Federation (ESGF) ready to use for climate and climate impact studies for Europe.

2 Resources

Simulation results (2994 TB) are currently stored in the HPSS and were post-processed to enable their publication in the ESGF, which "is an international collaboration for the software that powers most global climate change research, notably assessments by the Intergovernmental Panel on Climate Change (IPCC)" (https://esgf.llnl.gov/mission.html). Data stored in made available via the ESGF requires a special data format (CMOR), therefore all data from WRF had to be post-processed. For post-processing of the five-year time-slices 160 TB Workspaces are needed to reformat the raw model-output (8 GB/day) for the analyses in EURO-CORDEX and long-term storage in the ESGF climate data base for worldwide users. For post-processing shell scripts applying climate data operators and Fortran routines are applied. The latter are able to CMORize the WRF-output data to the required ESGF format. Computing power of 1000 node hours (Rth) to process 2*50 years of hourly output for 85 variables were needed. Table 1 summarizes the resources for the reporting period.

3 Results

The postprocessed data is analysed in current and future projects funded by the BMBF and DFG, e.g. in the project "Investigation of an Ensemble of Regional Climate Models with Respect to Land-Atmosphere Feedback and Extreme Precipi-

Table 1 Storage and computing resources used for WRFCLIM from 03/2019−02/2020

Workspace	160 TB
Node hours for computing	1484 Rth
Node hours for storage online accounting	1103715 Rth
HPSS storage space	2994 TB

tation (LAFEP)" was approved and will be funded from March 2020 by the BMBF within the Research and Development Project "Klimawandel und Extremereignisse (ClimXtreme)", which is scheduled for funding from March 2020 to February 2023. Recent analyses including the WRFCLIM project simulations are published in international peer-reviewed journals and under review within the EURO-CORDEX model ensemble analyses [1–5, 7, 8]. Further the data is published in the ESGF data base (e.g. https://esgf-data.dkrz.de/pqrojects/esgf-dkrz/) for climate and climate impact research and to be accessible via the climate information platform (https://freva.met.fu-berlin.de/).

4 Outlook

The analysis of this new ensemble of regional climate simulations with respect to intensity, frequency, and duration of heavy precipitation events (HPEs) will be the upcoming focus of WRFCLIM within the BMBF funded Research and Development Project ClimXtreme. Particularly, we focus on the role of Land-Atmosphere (L-A) feedbacks for the formation of these events. In order to realize this, the project concentrates on the following four major objectives:

1. Identification and verification of HPEs in form of single precipitation events and intensive precipitation episodes in an ensemble of regional climate simulations.
2. Determination and evaluation of process-related physical quantities and forcing mechanisms relevant for the formation and duration of the HPEs focusing on metrics characterizing L-A feedback.
3. Quantification of past and future changes of HPEs and related L-A feedbacks.
4. Investigation of an improved representation of L-A feedback mechanisms and extreme precipitation events in CP simulations on the regional and global scale.

These objectives will be addressed analyzing the 3-hourly data from the available EURO-CORDEX and ReKliEs-De Ensemble members, namely the WRFCLIM simulations and identifying episodes with HPE events in the past and future. Four of the episodes will be selected for downscaling WRFCLIM EURO-CORDEX ensemble members with WRF. This will be complemented with three global CP simulations covering northern hemisphere spring and summer with the Model for Prediction Across Scales (MPAS). We expect new insight in the role of dynamical and thermodynamical processes and their feedback on the forcing mechanisms leading to heavy precipitation as well as their expected modifications due to climate change. By means of new CP simulations, the expected improvement of precipitation statistics and L-A feedback metrics will be studied by systematic medium-term episode simulations of HPEs.

The improved understanding of HPEs and the quantification of their potential future changes provide a useful input for research institutions involved with climate impact studies, environmental agencies, and other application-oriented institutions

for the development and improvement of adaptation strategies. This work will also be imbedded in ongoing projects of the WCRP such as the CORDEX Flagship pilot studies Land Use and Climate Across Scales (LUCAS) and FPS Convection.

Acknowledgements Computational Resources for the model simulations on the CRAY XC40 within WRFCLIM were kindly provided by HLRS. We thank HLRS staff for their valuable support.

References

1. N. Ban, C. Caillaud, E. Coppola, E. Pichelli, S. Sobolowski et al., The first multi-model ensemble of regional climate simulations at kilometer-scale resolution, part I: evaluation of precipitation. *Climate Dynamics* (2021). https://doi.org/10.1007/s00382-021-05708-w
2. K. Bülow, H. Huebener, K. Keuler, C. Menz, S. Pfeifer, H. Ramthun, A. Spekat, C. Steger, C. Teichmann, K. Warrach-Sagi, User tailored results of a regional climate model ensemble to plan adaption to the changing climate in Germany. Adv. Sci. Res. **16**, 241–249 (2019). https://doi.org/10.5194/asr-16-241-2019
3. J.H. Christensen, M.A.D. Larsen, O.B. Christensen, M. Drews, M. Stendel, Robustness of European climate projections from dynamical downscaling. Clim. Dyn. **53**, 4857–4869 (2019). https://doi.org/10.1007/s00382-019-04831-z
4. D. Jacob, C. Teichmann, S. Sobolowski, E. Katragkou, I. Anders, M. Belda, R. Benestad, F. Boberg, E. Buonomo, R.M. Cardoso, A. Casanueva, O.B. Christensen, J.H. Christensen, E. Coppola, L. de Cruz, E.L. Davin, A. Dobler, M. Domínguez, R. Fealy, J. Fernandez, M.A. Gaertner, M. García-Díez, F. Giorgi, A. Gobiet, K. Goergen, J.J. Gómez-Navarro, J.J.G. Alemán, C. Gutiérrez, J.M. Gutiérrez, I. Güttler, A. Haensler, T. Halenka, S. Jerez, P. Jiménez-Guerrero, R. Jones, K. Keuler, E. lström, S. Knist, S. Kotlarski, D. Maraun, E. van Meijgaard, P. Mercogliano, J.P. Montávez, A. Navarra, G. Nikulin, N. de Noblet-Ducoudré, H.J. Panitz, S. Pfeifer, M. Piazza, E. Pichelli, J.P. Pietikäinen, A.F. Prein, S. Preuschmann, D. Rechid, B. Rockel, R. Romera, E. Sánchez, K. Sieck, P.M. Soares, S. Somot, L. Srnec, S.L. Sørland, P. Termonia, H. Truhetz, R. Vautard, K. Warrach-Sagi, V. Wulfmeyer, Regional climate downscaling over Europe: perspectives from the EURO-CORDEX community. Reg. Environ. Change **20**(51) (2020). https://doi.org/10.1007/s10113-020-01606-9
5. A. Lavin-Gullon, J. Fernandez, S. Bastin, R.M. Cardoso, L. Fita, T.M. Giannaros, K. Goergen, J.M. Gutierrez, S. Kartsios, E. Katragkou, T. Lorenz, J. Milovac, P.M.M. Soares, S. Sobolowski, K. Warrach-Sagi, Internal variability versus multi-physics uncertainty in a regional climate model. Int. J. Climatol. (2020). https://doi.org/10.1002/joc.6717
6. C. Skamarock, B. Klemp, J. Dudhia, D. Gill, D. Barker, G. Duda, X. Huang, W. Wang, G. Powers, A Description of the Advanced Research WRF Model Version 3. Technical report, NCAR, Boulder/CO, USA (2008). https://doi.org/10.5065/D68S4MVH
7. R. Vautard, N. Kadygrov, C. Iles, F. Boberg, E. Buonomo, K. Bülow, E. Coppola, L. Corre, E. van Meijgaard, R. Nogherotto, M. Sandstad, C. Schwingshackl, S. Somot, E. Aalbers, O.B. Christensen, J.M. Ciarlò, M.E. Demory, F. Giorgi, D. Jacob, R.G. Jones, K. Keuler, E. Kjellström, G. Lenderink, G. Levavasseur, G. Nikulin, J. Sillmann, C. Solidoro, S.L. Sørland, C. Steger, C. Teichmann, K. Warrach-Sagi, V, Wulfmeyer, Evaluation of the large EURO-CORDEX regional climate model ensemble. J. Geophys. Res. Atmos. (2020). https://doi.org/10.1029/2019JD032344
8. M. von Cossel, M. Wagner, J. Lask, E. Magenau, A. Bauerle, V. von Cossel, K. Warrach–Sagi, B. Elbersen, I. Staritsky, M. van Eupen, Y. Iqbal, N. Jablonowski, S. Happe, A. Fernando, D. Scordia, S. Cosentino, V. Wulfmeyer, I. Lewandowski, B. Winkler, Prospects of bioenergy cropping systems for a more social–ecologically sound bioeconomy. Agronomy **9**(10) (2019). https://doi.org/10.3390/agronomy9100605

WRF Simulations to Investigate Processes Across Scales (WRFSCALE)

Hans-Stefan Bauer, Thomas Schwitalla, Oliver Branch, and Rohith Thundathil

Abstract Different scientific aspects ranging from boundary layer research and air quality modeling to data assimilation applications were addressed with the Weather Research and Forecasting (WRF) model from the km-scale down to the turbulence-permitting scale.

Case study simulations in as different regions as the central United States and the United Arab Emirates were performed to investigate the evolution of the convective boundary layer. The multi-nested WRF setup, driven by the operational analysis of the European Centre for Medium-range Weather Forecasts (ECMWF), high-resolution terrain, and land cover data sets simulated a realistic evolution of the internal turbulent structure of the boundary layer including the transitions between daytime and nighttime conditions. Simulations with km-scale resolution over the United Arab Emirates revealed the performance of the WRF model compared to surface station data in this arid region.

An air quality forecast system based on the WRF-Chem model was set up and its performance was tested with a resolution as fine as 50 m for Stuttgart. It demostrated good performance in representing the morning an evening rush hour peak concentrations and their reduction due to the developing daytime turbulence.

Data assimilation experiments demonstrated the beneficial influence of state-of-the-art lidar measurements on the forecast performance of WRF. A further improvement was found when the more sophisticated hybrid 3DVAR-ETKF method was

H.-S. Bauer (✉) · T. Schwitalla · O. Branch · R. Thundathil
Institute of Physics and Meteorology, Garbenstrasse 30, 70599 Stuttgart, Germany
e-mail: hans-stefan.bauer@uni-hohenheim.de

T. Schwitalla
e-mail: thomas.schwitalla@uni-hohenheim.de

O. Branch
e-mail: oliver_branch@uni-hohenheim.de

R. Thundathil
e-mail: rohith.thundathil@uni-hohenheim.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_31

applied, since this method includes a more sophisticated flow-dependent model error contribution spreading the information of the observations more realistically in the domain.

1 Introduction and Motivation

Numerical models have the potential to address the inherent weaknesses of observation-based approaches, since they can provide a full 4D representation of the atmosphere and produce a consistent state with respect to the 3D thermodynamic atmospheric fields, cloud water, cloud ice, and diagnostic variables such as precipitation. Therefore, numerical models are excellent tools to improve our understanding of atmospheric processes across scales.

Recent results have demonstrated that grid resolutions of less than around 4 km are necessary for a realistic representation of mesoscale processes, especially in terms of the spatial and temporal distribution of precipitation. Further improvements are expected if a chain of grid refinements is performed down to the Large-Eddy Simulation (LES) scale (100 m and below), as further details of land-surface atmosphere (LSA) interaction are resolved (e.g. [1]). Through application of extremely high resolution and low-pass filtering, larger eddies and the dominant spectra for turbulent transport of heat and moisture can be simulated explicitly.

The WRF model system, described with more detail in earlier reports, provides the opportunity to perform LES simulations under realistic conditions because of the wide range of scales it can be applied over. Using a nesting strategy that covers the synoptic weather situation and the mesoscale circulations, in combination with data assimilation, will ensure a forcing of the LES that is as close as possible to the real weather situation.

2 Work Done Since March 2019

2.1 LES Simulations to Better Understand Boundary Layer Evolution and High-Himpact Weather (LES-PROC)

The Land Atmosphere Feedback Experiment (LAFE) took place in August 2017 at the Southern Great Plains site of the Atmospheric Radiation Measurement Program (ARM) in Oklahoma. Many different instruments were brought to the site and the measurement strategy was optimized in a way to derive as much information as possible. More details about the campaign and the applied measurement strategy can be found in [18].

We so far focused our simulations on one LAFE case on the 23rd of August 2017, one of the golden measurement days during the campaign. It was a clear sky day where the operation of the lidar systems were temporally extended to include the evening transition of the convective boundary layer.

The simulations were started at 06 UTC (01 a.m. local time) and run for 24 h within the 24 h wall time limit. WRF was set-up in two different configurations.

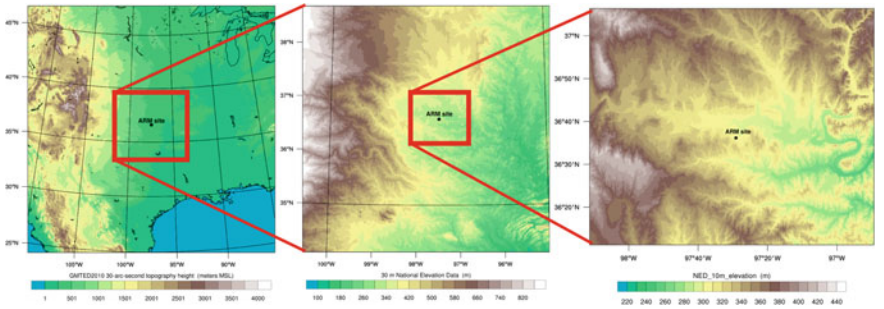


Fig. 1 Domain configuration of the first LAFE simulation. From left to right the domains with 2500 m, 500 m and 100 m resolution.

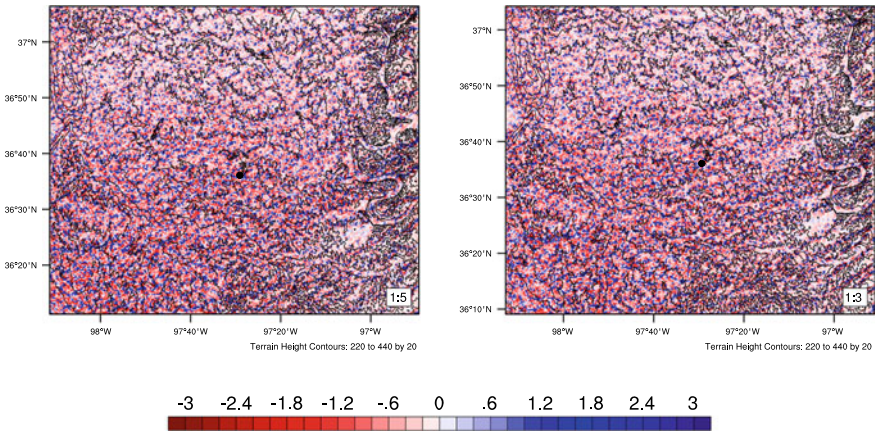


Fig. 2 Vertical velocity 1000 m above sea level at 2 p.m. local time, 23 August 2017 in the innermost 100 m domain as simulated by two different domain configurations with 2500 m—500 m—100 m (left) and 2700 m—900 m—300 m—100 m (right).

One using three domains with 2500 m, 500 m and 100 m resolution and a 4-domain setup with 2700 m, 900 m, 300 m and 100 m. The outer domain was driven by the operational analysis of the European Centre of Medium Range Weather Forecasting (ECMWF). The size of the domains is 1000×1000 grid cells in the outer domains and 1201×1001 grid cells in the innermost 100 m domain. The vertical resolution is the same in all domains with 100 vertical levels up to a height of 50 hPa. Figure 1 illustrates the 3-domain setup. The location of the ARM site is marked in the different domains.

Figure 2 compares the representation of turbulence in the two different domain setups. The innermost 100 m domains have the same size and cover the same area in both configurations.

Although differences in the representation of turbulence occur, the breakup into turbulent eddies is similarly simulated by both configurations. Even the flow adjust-

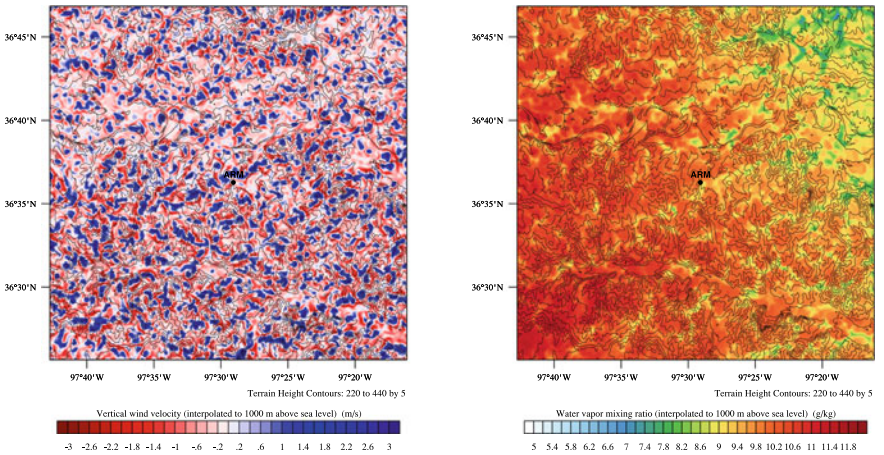


Fig. 3 Representation of the convective boundary layer zoomed into a small region around the ARM SGP site at 2 p.m. on 23 August 2017. Vertical velocity $[ms^{-1}]$ (left) and water vapor mixing ratio $[gkg^{-1}]$ 1000 m above sea level.

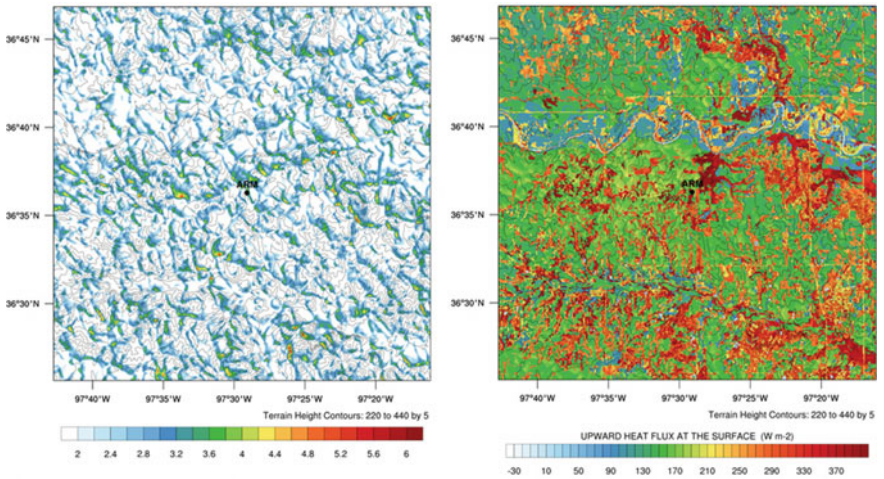


Fig. 4 Representation of the 10-m horizontal wind velocity $[ms^{-1}]$ and the surface sensible heat flux $[Wm^{-2}]$ at 2 p.m. on 23 August 2017.

ment along the upwind eastern and northern boundaries is similar, suggesting that a computationally more efficient grid ratio of five (instead of three) can be applied for future investigations.

The well-developed boundary layer is better seen when zooming into the innermost domain. Figure 3 shows the vertical velocity and water vapor mixing ratio fields at 1000 m a.s.l. in the early afternoon of 23 August 2017. The 10 m horizontal wind field and the surface sensible heat flux are shown in Fig. 4.

The size of the updraft plumes (blue) and the compensating downdrafts (red) are nicely captured and correspond in size to observations done in the region e.g. [17, 18].

The upward transport of near surface moist air by the updraft plumes is illustrated with the water vapor mixing ratio field (left panel of Fig. 3). At the surface distinct gust lines occur as consequence of the well-established turbulence as seen on the left panel of Fig. 4. The right panel of Fig. 4 shows the surface sensible heat flux. It strongly depends on the underlying land cover type and the soil characteristics with larger values over barren or grassland areas, medium values over cropland and lower values over forested areas in the river valley in the northern part of the domain.

The forecast length of 24 h allows the analysis of a full diurnal cycle of the evolution of the boundary layer. As an example, Fig. 5 shows time-height cross sections of the two horizontal wind components and the horizontal wind speed. Time series output at selected model grid points are written in addition to provide data in 10 s resolution.

The different stages of the development of the boundary layer are clearly seen. During night, a shallow nighttime boundary layer is seen, overlaid by the residual layer, namely the convective boundary layer of the previous day. In the morning local time, with the onset of turbulence, the new convective boundary layer is growing. The turbulent fluctuations, seen in all three panels, are typical for the development of an undisturbed convective boundary layer. In the evening after sunset, turbulence diminishes and finally a new nighttime stable boundary layer develops.

Another interesting feature, commonly found in this region is the low-level southerly jet, prominently shown by the lower panel of Fig. 5, transporting moist air from the Gulf of Mexico to the region.

2.2 Seasonal Land Surface Modification Simulations (UAE-1)

As part of the work to select interesting—1 km grid resolution—cases for the land surface modifications, we conducted a set of daily forecasts on Hazelhen at 2.7 km grid resolution from Jan 01 to Nov 30, 2015. The 0.025 resolution domain with 900(x) * 700(y) grid cells is shown in Fig. 6, b.

These grid resolutions of between 2–4 km are becoming typical for operational forecasting in the UAE. It is vital that these forecasts are of a high skill given the common occurrence of convective storms, flash floods, heat waves, and low-visibility events such as thick fog and dust storms. This near one-year dataset gives us a unique opportunity to evaluate the skill of this model to reproduce important surface quantities. Previous assessments of WRF in the UAE are valuable e.g. [3, 13], but to date no verification has been carried out in the UAE at such a high resolution for such a long time series. Such simulations are vital for robust statistics and seasonal assessments.

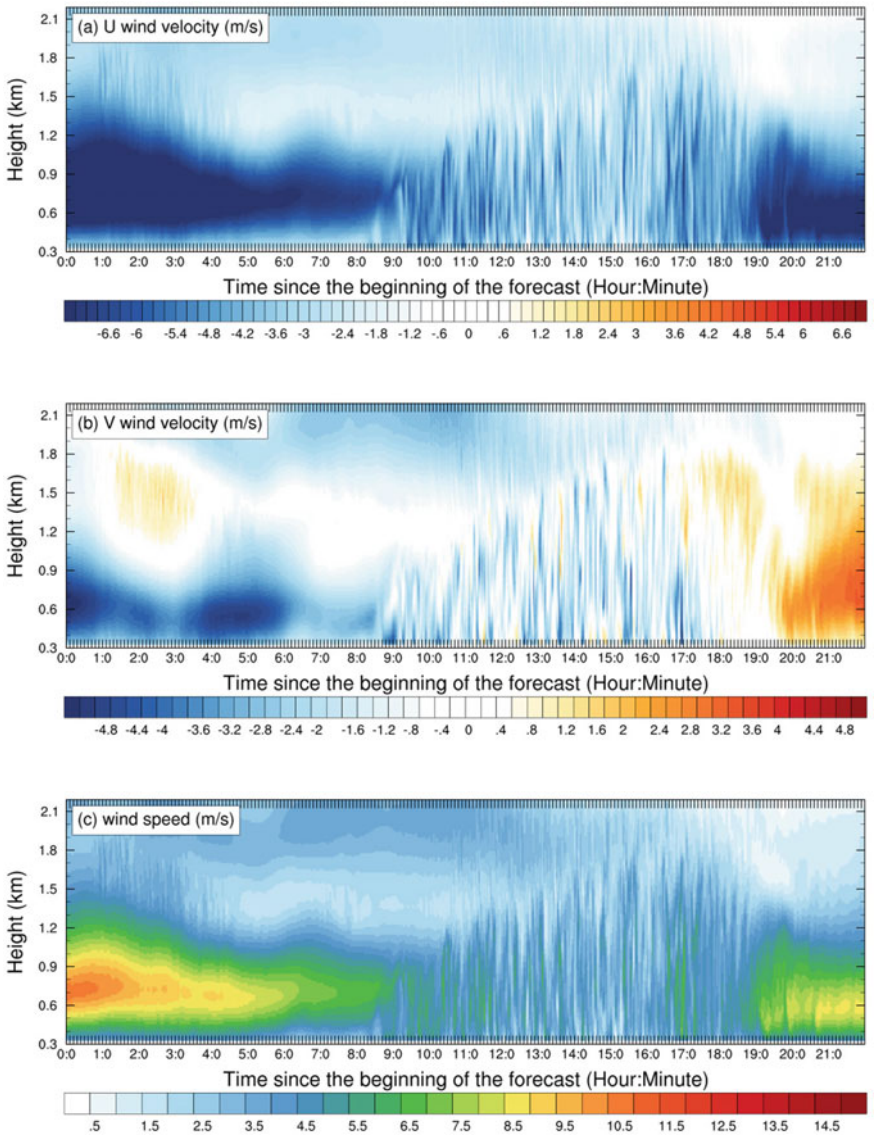


Fig. 5 Time-height cross sections of the west-east (top panel) and the south-north (middle panel) component of the wind velocity (ms-1). The bottom panel shows the total wind velocity calculated as the vector sum of the two components. The X-Axis marks the time in hours since the beginning of the forecast (00 corresponds to 06 UTC or 01 local time in Oklahoma).

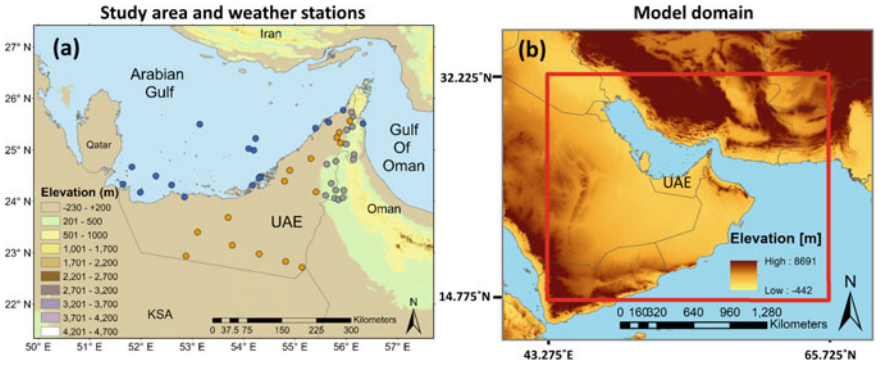


Fig. 6 The study area (left) showing the United Arab Emirates. Marked on this topography map are the 48 weather stations on which comparisons were made. These are split into 3 groups based on the geophysical region – grey dots, mountain – orange dots, desert – blue dots, marine.

We compared model hourly values for 2 m temperature (T-2m), 2 m dew point (TD_2m), and 10 windspeed (UV-10m) with those of 48 surface weather stations spread over the UAE (Fig. 7). Furthermore, we compared model performance within different times of the day, and in different sub regions of the UAE – mountain, marine, and desert. It is useful to do this to examine performance in regions where climate differs significantly. For instance, convection initiation is common in the Al Hajar mountains [2]. Mountain stations are located above 200 m a.s.l in the Al Hajar Mountains of eastern UAE. Marine stations are either on Islands or on land but within 5 km of the ocean. Desert stations are inland below 200 m.a.sl (see Fig. 6, a).

Results

Bias statistics provide a measure of the mean error including sign of the deviation of the model forecast compared with the observations.

Temperature biases in the daytime are quite low ($\leq 1\,^{\circ}C$), whereas there is a strong nighttime bias exceeding $2\,^{\circ}C$ (Fig. 7, a). This becomes quite extreme in the desert (exceeding $3\,^{\circ}C$). The reasons for this nighttime bias are not certain but its likely due to poor representation of the stable nocturnal boundary layer which can be very shallow. In general, performance is best within the marine region. This is likely due in part to an improvement provided by the daily sea surface temperature ingested into the model. It is notable that the marine temperatures are slightly underestimated during the day.

Dew point appears to be relatively well reproduced with biases generally $\leq 1\,^{\circ}C$ (Fig. 7, b). Nocturnal biases are high in the mountains however. Together with high temperature bias, it appears there is a problem with too much vertical mixing in the boundary layer. In this case the best performance is in the desert region with only slightly negative biases ($\leq 0.5\,^{\circ}C$).

Windspeeds are in general over-estimated by WRF, particularly during the day with biases reaching $2\,ms^{-1}$ (Fig. 7, c). This was also observed in [13]. Biases are

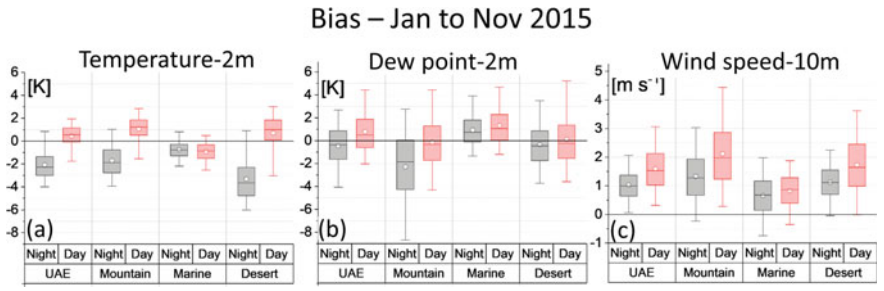


Fig. 7 Box plots of bias for 2 m air temperature, dew point and 10 m windspeed - panels (a–c) respectively for all time steps over the period of Jan-Nov 2015. Statistics are divided by region (UAE, Mountain, Marine, Desert) and then by nighttime and daytime hours (respectively, nighttime 18:00–05:00 (grey boxes) and daytime 06:00–17:00 (red boxes) in local time). On the box plots, the centre line represents the mean, the white circle is the median, box ends are 25% and 75% percentiles and the whiskers are 5% and 95% percentiles. Also marked is a zero-reference line.

strongest in the mountains (2 ms^{-1}), followed by the desert (1.5 ms^{-1}). The marine biases are lowest at 1 ms^{-1}. These wind speed biases are a matter of concern.

It is also important to assess the reproduction of the hourly diurnal cycle to assess model performance in more detail – in particular at important times such as the local noon, where radiation is strongest and at transition times when the daytime convective boundary layer develops at sunrise and then collapses at sunset. Models often have problems during these periods. Figure 8 shows mean summer diurnal cycles along with the standard deviation for WRF and the observations. From these plots we can also look at the regional differences in climate.

Temperature curves show the marked differences in regional mean temperature with the highest maximum T-2m in the desert (318 K or 45 °C) and lowest in the marine region (312 K or 39 °C). They also reflect the results of bias plots but in more detail. We can see that the variability across the UAE is quite low and the model reproduces this well. This low variability indicates that if temperatures are high, then they are high everywhere. The nighttime biases are pronounced, but we can see that the performance at transition times (03–04:00 UTC and 14–15:00 UTC) is actually rather good with the lowest biases occurring here. The problem periods are clearly in the middle of the night and at local noon.

Dew point curves highlight much greater variability, which can be expected given the often highly heterogeneous nature of moisture in the air. We can see that the marine region is much more humid as expected. In general, the performance is good in the daytime but with slight wet biases in the marine and desert regions. There is a marked dry nocturnal bias in the mountains. Reiterating the likely problem of too much mixing.

Windspeed curves show high variability particularly during daytime in the mountains and desert (± 2 ms^{-1}) where the biases are highest (up to 2 ms^{-1}). The evening transition time exhibits the best performance.

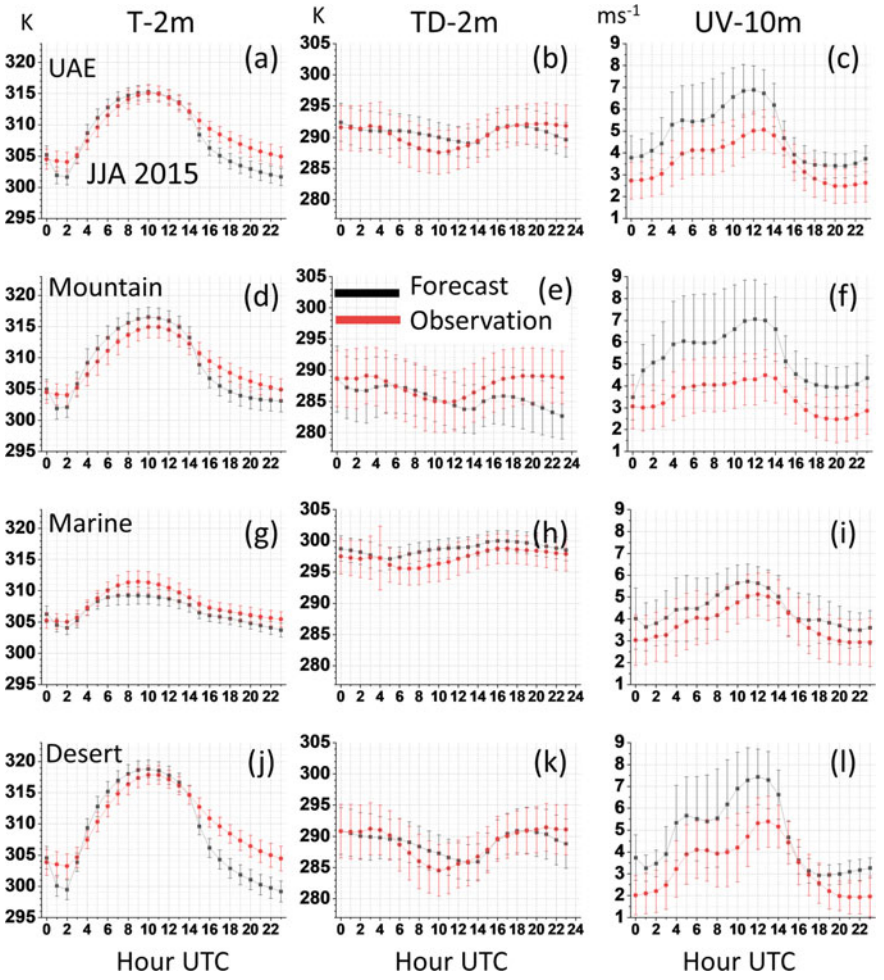


Fig. 8 Diurnal cycles of mean summer (JJA) WRF forecasts (black) compared to observations (red) within different regions. In each plot the mean and standard deviation is shown. Variables are temperature, dew point, and windspeed.

In general, there is an apparent relationship between temperature bias and windspeed bias, whereas TD-2m biases appear to be more independent. The only exception to this is during the night, when T-2m and TD-2m biases do appear linked (Fig. 7, a and b), likely due to difficulties in simulation of the stable PBL.

The aim of this study was to assess the skill of WRF-NOAHMP in reproducing surface quantities over the UAE; to identify regional, seasonal, and diurnal differences in performance; and estimate potential sources of model deficiencies. This study highlights the value of splitting the analysis temporally and spatially for verification. Although bulk scores and statistics for the whole UAE are useful, there are

regional, diurnal and seasonal compensating biases, revealing important information on how the model performance changes spatially and temporally.

Studies such as these are vital for accurate assessment of WRF nowcasting performance and to identify model deficiencies and areas for improvement. By combining sensitivity tests with process and observational studies [2, 3, 10, 11] with seasonal verification studies such as these, we should move towards improved forecasting systems for the UAE and other arid regions.

2.3 *Turbulence-Permitting Particulate Matter Forecast System (OpenFCST)*

The aim of the Open Forecast is to develop and set up a prototype for particulate matter forecasts over the Stuttgart Metropolitan area. The targeted resolution is the turbulence permitting scale, i.e. a horizontal resolution between 50–150 m. The size of the applied domains is determined by the idea to potentially apply the PMFS as an operational forecasting system. As the computational cost of running a turbulence permitting model domain including atmospheric chemistry is very high, the domain sizes are kept as small as possible but as large as necessary. The model domains for the PMFS have been set up and the WRF-Chem model [5, 14] was successfully compiled at HLRS.

Due to the target resolution of 50 m over the Stuttgart metropolitan area, we apply two outer coarser domains with 250 m and 1250 m resolution. The grid dimensions are 800×800, 601×601 and 601×601 cells. Figure 9 shows the innermost model domain.

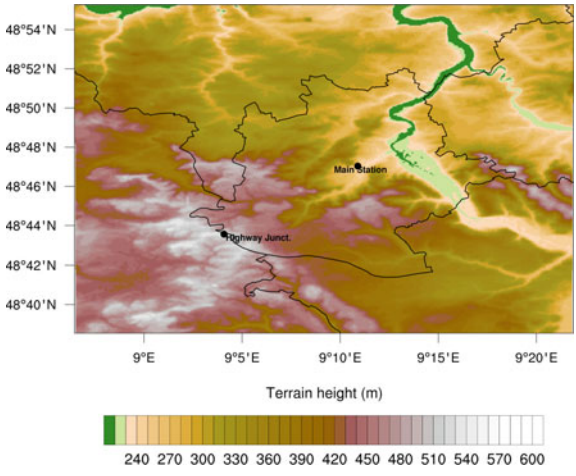


Fig. 9 Innermost model domain with 50 m horizontal resolution.

The communication between the domains is one-way, that is information from the parent domain is used as input for the child domain at the domain boundaries. There is no feedback from the child to the parent domain.

The vertical resolution is 100 levels up to 50 hPa (22 km above sea level) with a high density in the lowest 2000 m above the surface. Meteorological input data are obtained from the operational ECMWF[1] analysis available on 137 vertical levels. To further speed up the simulation, adaptive time stepping is applied. The maximum allowed time step for the outermost model domain is 15 s while it is limited to 3 s for the innermost model domain. In case stronger turbulence occurs, the time step is reduced to 0.5 s.

As the targeted horizontal resolution is very high, the so far available data set for terrain and land cover do not have a sufficient horizontal resolution. Therefore, land cover data for all domains is obtained from the Copernicus CLC 2012[2] data set which is aggregated to a grid with 100 m resolution. Depending on the applied resolution, terrain information is obtained from the Global Multi-resolution Terrain Elevation Data (GMTED2010[3]) data set as well as from Shuttle Radar Topography Mission (SRTM[4]). Additionally, we incorporated a new land-cover data set from Landesamt für Umwelt Baden-Württemberg (LUBW[5]), which is derived from LANDSAT[6] in 2010 and is available at 30 m resolution suitable for our target domain. The original LUBW categories were mapped to the IGBP-MODIS[7] categories required by the WRF model using the QGIS software package.

The following set of physics packages has been applied. Shortwave and longwave Radiation is parameterized by the RRTMG scheme [7], which interacts with the cloud microphysics and aerosols. The Noah-MP land-surface model (LSM) [12] calculates soil and surface fluxes as well as soil temperatures and soil moisture coupled to the surface layer parametrization. The outermost model domain applies the YSU planetary boundary layer (PBL) parametrization [6], while the other two domains resolve turbulence directly without any PBL scheme. Cloud microphysics is simulated by the Thompson 2-moment cloud microphysics [15] propagating hydrometeors and its number concentrations. No cumulus parametrization is applied, as the model can explicitly resolve convection.

The Regional Acid Deposition Model, 2nd generation (RADM2), parameterizes the atmospheric chemistry in all domains. RADM2 features 21 inorganic and 42 chemical species including more than 100 chemical reactions. Aerosols are represented by the Modal Aerosol Dynamics Model for Europe (MADE) and Secondary Organic Aerosol Model (SORGAM) scheme considering size distributions, nucleation, coagulation, and condensational growth. The scheme is called every 2 min for

[1] https://www.ecmwf.int.

[2] https://land.copernicus.eu/pan-european/corine-land-cover/clc-2012.

[3] https://www.usgs.gov/land-resources/eros/coastal-changes-and-impacts/gmted2010.

[4] https://www2.jpl.nasa.gov/srtm/.

[5] https://www.lubw.baden-wuerttemberg.de/.

[6] https://landsat.gsfc.nasa.gov.

[7] https://icdc.cen.uni-hamburg.de/daten/land/modis-landsurfacetype.html.

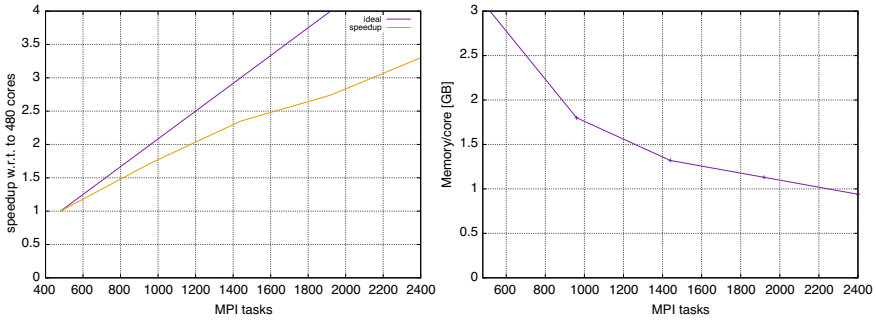


Fig. 10 Speedup of the WRF-CHEM simulation with respect to 480 cores (left) and memory consumption/core (right).

the outer domain and every 45 s in the inner domains. Due to the applied solver, this option allows for a larger time step for the chemistry module compared to other even more simple schemes.

Recent results indicate that the application of an urban canopy model [8] improves the forecast quality especially in terms of surface fluxes. This requires special data categories for low and high residential areas as well as for industrial areas that are available in the CLC 2012 and LUBW data sets.

In order to investigate the performance of the model system, basic scaling test have been performed on the XC 40 system at HLRS. The left panel of Fig. 10 exemplarily displays the scaling of the WRF-CHEM model with respect to 20 nodes for the smallest, innermost domain. Less than 20 nodes are not possible, as the memory requirements per core would exceed the available memory of 128 GB per node.

It is seen, that the scaling is not linear, and it appears that MPI communication becomes the driving factor when applying more than 80 nodes. Compared to a pure WRF simulation, the required wall time per time step in our configuration is around 4–5 times higher. Due to the high resolution and thus small model integration time steps, a 24 h forecast currently requires 40 h wall clock time. The total data amount for a 24 h forecast using 5 min output intervals including 40 chemical constituents is about five TB for the innermost model domain. Currently, the hybrid mechanism of MPI and OPENMP cannot be used for WRF-Chem as this is not tested and would in addition exceed the capabilities during the Open Forecast project.

To initialize the model and provide chemical boundary conditions, data from the Whole Atmosphere Community Climate Model (WACCM; [9]) are applied using the MOZBC conversion tool. Recently, high-resolution emission data for Europe from the Copernicus Atmosphere Monitoring Service (CAMS1;[8] [4]) became available. The resolution is approx. 7×7 km and is based on annual emissions of 2016. This product provides emissions of $PM10$, $PM2.5$, SO_2, CO, NO, NO_2, and CH_4. As the data cannot be ingested directly with the available procedures, the data conversion

[8] https://atmosphere.copernicus.eu

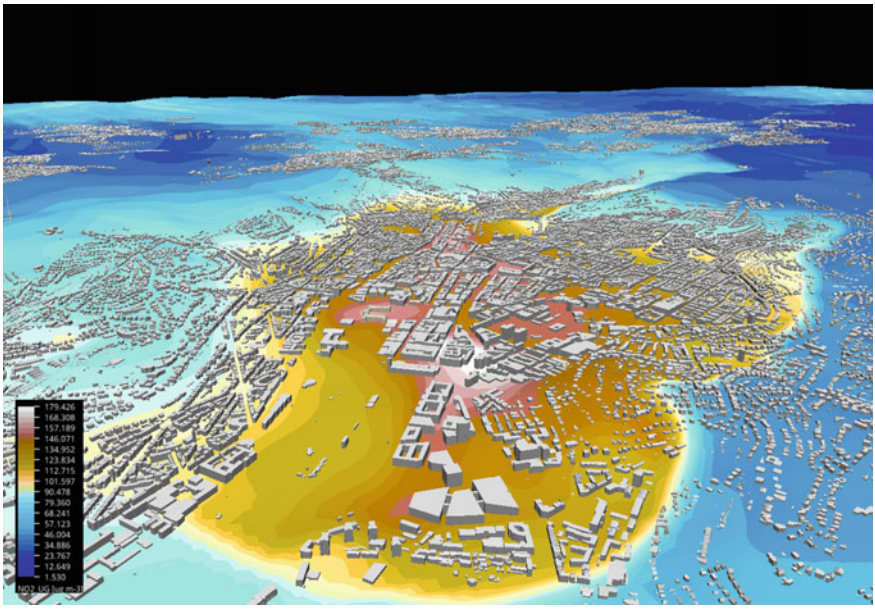


Fig. 11 Simulated NO_2 concentrations $[\mu gm^{-3}]$ during the morning peak traffic time on January 21, 2019. The viewing direction is from northeast to southwest with the Stuttgart main station in the foreground. Brownish colors indicate very high concentrations (Image courtesy: Leyla Kern, HLRS).

to the WRF-Chem data format was performed separately with the Earth System Model Framework (ESMF) interpolation utilities.

Figure 11 shows an example of the simulated NO_2 concentration over downtown Stuttgart during the morning peak traffic time. The higher concentrations in the Stuttgart basin can be clearly seen.

Another important aspect for environmental protection is the $PM10$ concentration and is shown in Fig. 12. Here, the highest $PM10$ concentrations are observed west of Stuttgart main station.

2.4 Assimilation of Lidar Water Vapor Measurements (VAP-DA)

The Earth's atmosphere rapidly changes with the influence of various factors directly and indirectly, irrespective of the factors being natural or anthropogenic. This can be visualized when we consider the Earth as a system modelled in a numerical model with several crucial parameters, namely, temperature, moisture, wind, pressure etc. The causality relation of all the parameters are defined in a numerical weather pre-

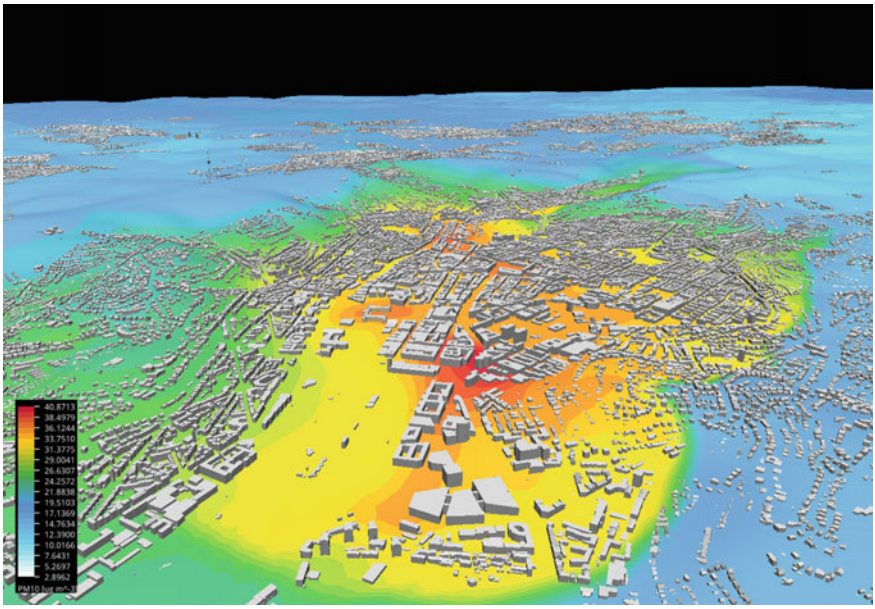


Fig. 12 Same as Fig. 11 but for $PM10$. Red colors denote high $PM10$ concentrations. (Image courtesy: Leyla Kern, HLRS).

diction model from micro- to macroscale. Estimation of these parameters in the most possible accuracy and resolution is important provided the resolution of measurements match the resolution of the NWP models. Temperature and moisture measurements play an important role in the atmospheric circulation, radiation and cloud formation. With high resolution instruments like lidars, estimation of these variables in their microscale resolution both temporally and spatially have become a reality. With these measurements, we can now understand the rapid changes in the planetary boundary layer where most of the convection initiation processes commence in microscale leading to macroscale developments.

Lidar data assimilation has a huge potential in the field of NWP in the future especially for regional weather forecasting. All prognostic equations in the NWP models are initial value problems where the final value depends on the initial state through the assimilation of lidar data, we can initiate the NWP model with the best possible quality measurements. Atmospheric profiles are currently provided by radiosondes which are not a continuous source of measurements and hence create sampling errors. Lidar data is a continuous source of information which include the dynamics of the atmosphere and lower sampling error. Hence lidar is a state-of-the-art instrument for future data assimilation efforts. Lack of a suitable forward operator for direct assimilation of water vapor mixing ratio was a significant limitation in the WRF NWP model. This is very important for the initiation of the model with real-time data which are less prone to external factors. Within WRFSCALE, a forward operator for

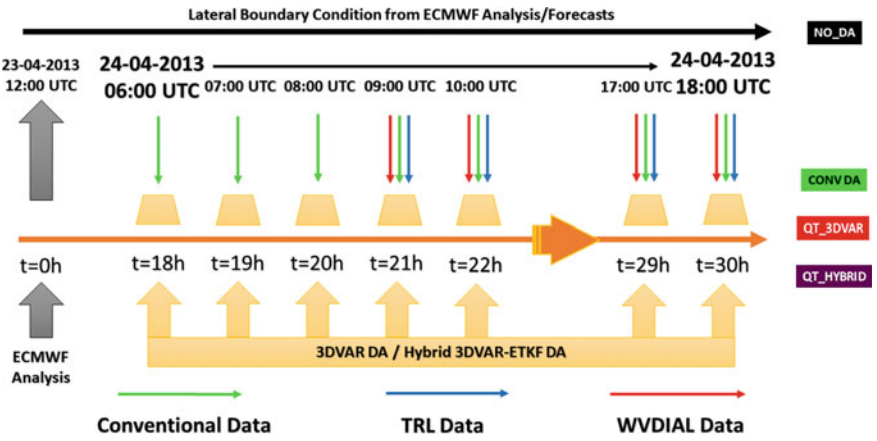


Fig. 13 The rapid update cycle and experiment flow diagram. 3DVAR: red and Hybrid: violet. The conventional DA without lidar DA is shown in green and no DA in black.

the direct assimilation of water vapor mixing ratio profiles has been accomplished with the modification of the atmospheric infrared sounding retrieval (AIRSRET) operator already available in the WRFDA system. We name it the thermodynamic lidar (TDLIDAR, [16]) operator.

Here we have used two different DA techniques to study the impact of lidar WVMR and temperature through the TDLIDAR operator. First, we studied the impact by the deterministic three-dimensional variational (3DVAR) data assimilation approach and secondly with the probabilistic hybrid 3DVAR-ensemble transform Kalman filter (ETKF) approach. The test case for the DA technique was the data obtained from one intensive observation period (IOP) of the HOPE campaign which was held in western Germany during the months of April and May 2013. A 10-hour dataset was assimilated into the model with hourly data inputs. A rapid update cycle data assimilation (Fig. 13) was performed at hourly intervals for the entire European domain at a spatial resolution of 2.5 km and a vertical resolution of 100 levels up to 50 hPa. The ensemble approach incorporated flow-dependent forecast error covariances into the assimilation process which alleviated the effects of the static covariance of the 3DVAR assimilation system. The hybrid DA system starts with a set of currently 10 ensemble members each with the same configuration as the previous experiments with a spatial resolution of 2.5 km and vertical resolution of 100 levels.

The impact of the lidar data with different DA techniques is depicted through analysis increment plots in Fig. 14.

Hybrid DA has more impact over a large area compared to the 3DVAR DA because of perturbations of prognostic variables over a wider area in the ensemble members thus involving larger spread in the mean ensemble member. Thus, the analysis increment of the hybrid approach shows the impact over a larger area compared to 3DVAR.

The profile plots in Figs. 15 and 16 show the impact of the assimilation on the vertical profile at the first assimilation and after six assimilation cycles for WVMR

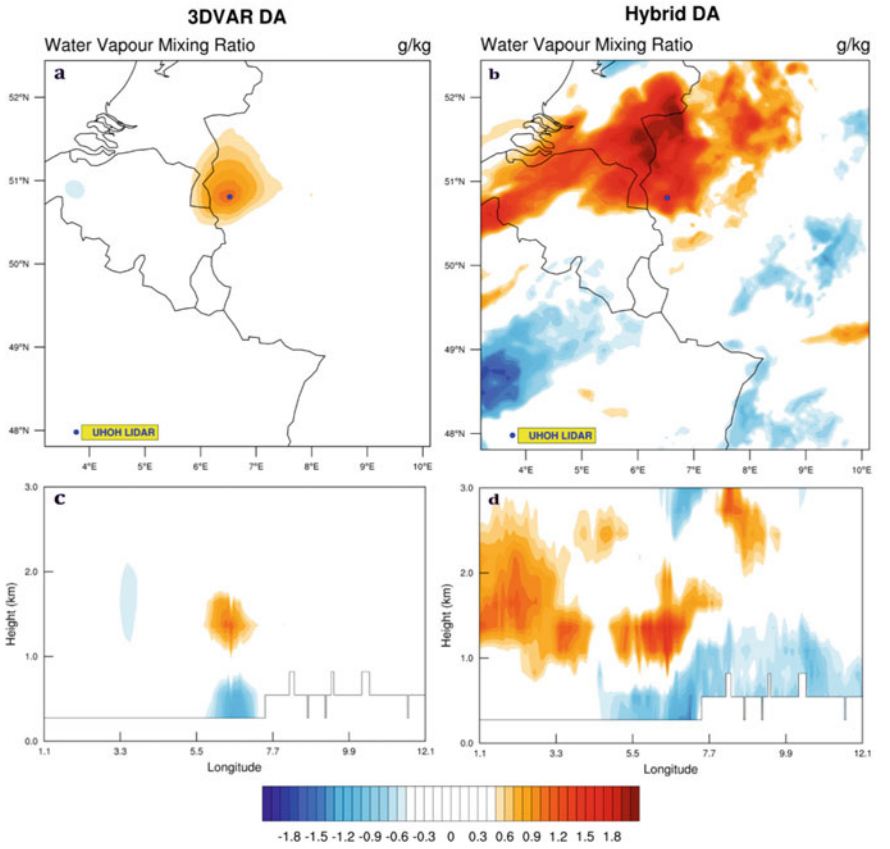


Fig. 14 Left panel: Water vapor mixing ratio analysis increment after 10 3DVAR DA cycles. Right panel: Same but after 10 Hybrid DA cycles. The blue dot represents the UHOH lidar location. a and b are spatial plots at 1200 m above ground level, and, c and d are longitude-height cross-section plots at the latitude the lidar was located.

and temperature. The QT_3DVAR, in Fig. 15, at 15 UTC assimilation shows good agreement with the DIAL profile although the model couldn't capture the boundary layer gradient in the first assimilation at 9 UTC. However, the boundary layer issue is solved by QT_HYBRID. Similarly, the temperature profile in Fig. 16, QT_3DVAR showed an unusual warming which was visible by overestimated temperatures. The dependence of static error covariance in the 3DVAR system created this issue which was alleviated in the Hybrid system.

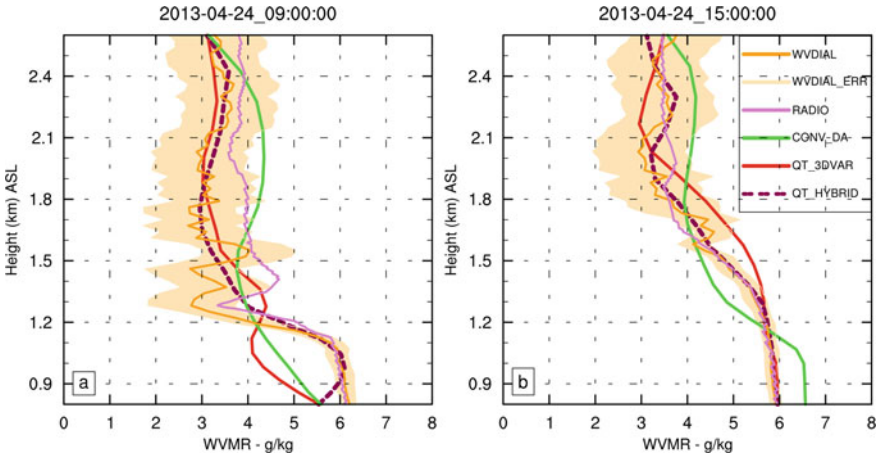


Fig. 15 Left panel: Water vapor mixing ratio profile during the first assimilation. Right panel: Water vapor mixing ratio profile at 15 UTC after 6 assimilation cycles.

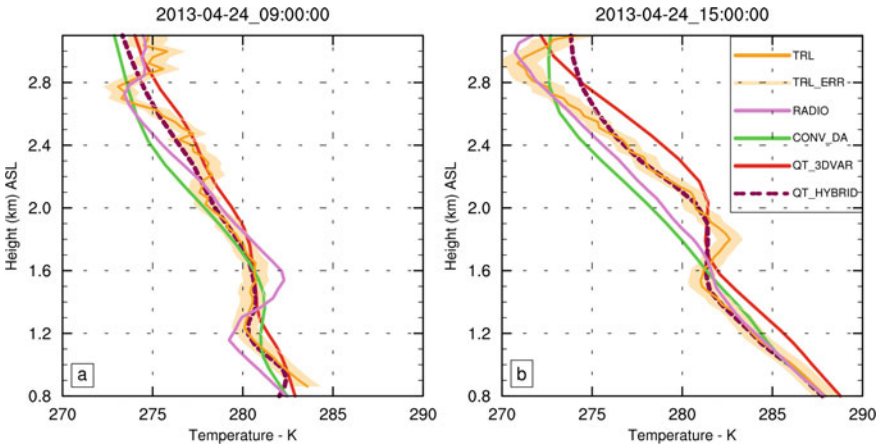


Fig. 16 Left panel: Temperature profile during the first assimilation. Right panel: Temperature profile at 15 UTC after 6 assimilation cycles.

References

1. H.S. Bauer, S.K. Muppa, V. Wulfmeyer, K. Warrach-Sagi, F. Späth, Multi-nested WRF simulations for studying planetary boundary layer processes on the turbulence-permitting scale in a realistic mesoscale environment. Tellus A **72**, 1–28 (2020)
2. O. Branch, A. Behrendt, Z. Gong, T. Schwitalla, V. Wulfmeyer, Convection initiation over the eastern Arabian peninsula. Meteorol. Zeitschrift **29**, 67–77 (2020)
3. R. Fonseca, M. Temimi, S. Thota, N.R. Nelli, M.J. Weston, K. Suzuki, J. Uchida, K.N. Kumar, O. Branch, Y. Wehbe, T. Al Hosari, N. Al Shamsi, A. Shalaby, On the analysis of the performance of WRFand NICAMin a hyperarid environment. Weather Forecast. **35**, 891–919 (2020)

4. C. Granier, S. Darras, H. Denier van der Gon, J. Doubalova, N. Elguindi, B. Galle, M. Gauss, M. Guevara, J.P. Jalkanen, J. Kuenen, C. Liousse, B. Quack, D. Simpson, K. Sindelarova, The Copernicus Atmospheric Monitoring Service global and regional emissions (April 2019 version). https://doi.org/10.24380/d0bn-kx16
5. G.A. Grell, S.E. Peckham, R. Schmitz, S.A. McKeen, G. Frost, W.C. Skamarock, B. Eder, Fully coupled online chemistry within the WRF model. Atmos. Environ. **39**, 6957–6975 (2005)
6. S.Y. Hong, Y. Noh, J. Dudhia, A new vertical diffusion package with an explicit treatment of entrainment processes. Mon. Wea. Rev. **134**, 2318–2341 (2006)
7. M.J. Iacono, J.S. Delamere, E.J. Mlawer, M.W. Shephard, S.A. Clough, W.D. Collins, Radiative forcing by long-lived greenhouse gases: calculations with the AER radiative transfer model. J. Geophys. Res. **113**(D13), 103 (2008)
8. H. Kusaka, F. Kimura, Coupling a single-layer urban canopy model with a simple atmospheric model.: Impact on urban heat island simulations for an idealized case. J. Meteor. Soc. Japan **82**, 67–80 (2004)
9. D.R. Marsh, M.J. Mills, D.E. Kinnison, J. Lamarque, N. Calvo, L.M. Polvani, Climate change from 1850 to 2005 simulated in CESM1(WACCM). J. Climate **26**, 7372–7391 (2013)
10. N.R. Nelli, M. Temimi, R.M. Fonseca, M.J. Weston, M.S. Thota, V.K. Vallapil, O. Branch, H.D. Wizemann, V. Wulfmeyer, Y. Wehbe, Micrometeorological measurements in an arid environment. Diurnal characteristics and surface energy balance closure. Atmos. Env. **234**, 104 745 (2020). https://doi.org/10.1016/j.atmospheres.2019.104745
11. N.R. Nelli, M. Temimi, R.M. Fonseca, M.J. Weston, M.S. Thota, V.K. Vallapil, O. Branch, V. Wulfmeyer, Y. Wehbe, T. Al Hosary, A. Shalaby, N. Al Shamsi, H. Al Naqbi, Impact of roughness length on WRF simulated land-atmosphere interactions over a hyper-arid region. Earth Sp. Sci. (2020). https://doi.org/10.1029/2020EA001165
12. G.Y. Niu, Z.L. Yang, K.E. Mitchell, F. Chen, M.B. Ek, M. Barlage, A. Kumar, K. LManning, D. Niyogi, E. Rosero, M. Tewari, Y. Xia, The community Noah land surface model with multiparameterization options (Noah-MP: 1. Model description and evaluation with local-scale measurements. J. Geophys. Res. **116**, D12 109 (2011)
13. T. Schwitalla, O. Branch, V. Wulfmeyer, Sensitivity study of the planetary boundary layer and microphysical schemes to the initialization of convection over the Arabian peninsula. Quart. J. Roy. Meteorol. Soc. **146**, 846–869 (2020)
14. W.C. Skamarock, J.B. Klemp, J. Dudhia, D.O. Gill, Z. Liu, J. Berner, W. Wang, J.G. Powers, M.G. Duda, D.M. Barker, X.Y. Huang, A description OD the Advanced Research WRF version 4 (2019). https://doi.org/10.5065/1dhf-6p97
15. G.P. Thompson, R. Field, R.M. Rasmussen, W.D. Hall, Explicit forecasts of winter precipitation using an improved bulk microphysics scheme. Part II: Implementation of a new snow parameterization. Mon. Wea. Rev. **136**, 5095–5115 (2008)
16. R. Thundathil, T. Schwitalla, A. Behrendt, S.K. Muppa, S. Adam, V. Wulfmeyer, Assimilation of lidar water vapor mixing ratio and temperature profiles into a coonvection-permitting model. J. Meteorol. Soc. Japan **98**, 959–986 (2020). https://doi.org/10.2151/jmsj.2020049
17. D.D. Turner, V. Wulfmeyer, L.K. Berg, J.H. Schween, Water vapor turbulence profiles in stationary continental convective mixed layers. J. Geophys. Res. **119**, 11 151–11 165 (2014). https://doi.org/10.1002/2014D022202
18. V. Wulfmeyer, D.D. Turner, B. Baker, A. Behrendt, T. Bonin, A. Brewer, M. Burban, A. Choukulkar, E. Dumas, R. Hardesty, T. Heus, J. Ingwersen, D. Lange, T. Lee, S. Metzendorf, S. Muppa, T. Meyers, R. Newsom, M. Osman, S. Raasch, A new research approach for obswerving and characterizing land-atmosphere feedback. Bull. Am. Meteorol. Soc. **99**, 1639–1667 (2018)

The Role of Lateral Terrestrial Water Flow on Land-Atmospheric Water Pathways

Joël Arnault and Harald Kunstmann

1 State of the Art

Numerical atmospheric models generally consider terrestrial hydrological processes as only being vertical, in order to estimate the surface heat fluxes for constraining the atmospheric lower boundary condition. This is for example the case for the Weather Research and Forecasting model (WRF, Skamarock and Klemp [25]) coupled with the Noah land surface model (LSM) with multi-parameterization options (Noah-MP, Niu et al. [18]). In this approach the lateral redistribution of soil moisture according to the topography and groundwater depth is, however, neglected. In order to relax this constraint and better represent soil moisture spatial variability, coupled atmospheric-hydrological models have been developed in recent years (e.g. [1, 14–16, 22, 24, 27]). In particular, the hydrologically-enhanced WRF-Hydro model allows to consider the coupling between overland flow, subsurface flow, soil moisture and atmospheric processes [9].

Several studies showed that the representation of terrestrial water flow in coupled atmospheric-hydrological models impacts the surface fluxes and planetary boundary

This project is part of the following scientific project:
• the DFG (German Research Foundation) Collaborative Research Center (CRC) 165/1 Wave to Weather (W2W) project A5 "The role of soil moisture and surface- and subsurface water flows on predictability of convection", supervised by Prof Dr Kunstmann and funded for the period 07/2015–06/2019.

J. Arnault (✉) · H. Kunstmann
Karlsruhe Institute of Technology, Institute of Meteorology and Climate Research, Kreuzeckbahnstr.19, 82467 Garmisch-Partenkirchen, Germany
e-mail: joel.arnault@kit.edu

H. Kunstmann
e-mail: harald.kunstmann@kit.edu

H. Kunstmann
University of Augsburg, Institute of Geography, Alter Postweg 119, 86159 Augsburg, Germany

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_32

layer dynamics, furthermore potentially influencing precipitation (e.g. [2, 4, 13–15, 22–24, 27, 29]). For the case of Central Europe, Arnault et al. [4] showed that the consideration of lateral terrestrial water flow in WRF-Hydro noticeably increases the normalized ensemble spread of daily precipitation where topography is moderate, surface flux spatial variability is high, and the weather regime is dominated by local processes.

2 Research Accomplished with the ForHLR2 Computing Resource

2.1 Overview

The general goal of this project is to further understand the role of lateral terrestrial water flow on land-atmospheric water pathways, with a focus on precipitation. In particular, we assess the role of lateral terrestrial water flow at continental scale for the European and West African continents using an ensemble modelling approach.

2.2 Impact of Lateral Terrestrial Flow On continental-Scale Precipitation and Dependency to Model Horizontal resolution (Project W2W-A5)

2.2.1 Original Research Plan

The description of lateral terrestrial water flow in WRF-Hydro generally increases the soil storage and the surface evaporation [2, 4, 23], and the contribution of surface evaporation to precipitation in a region increases with the size of the region (e.g. Trenberth [26], Arnault et al. [3]). Accordingly, the effect of lateral terrestrial water flow on simulated precipitation is expected to be maximized for a continental-scale simulation domain. Furthermore, in WRF-Hydro the lateral re-distribution of terrestrial water is considered on a subgrid, so that the effect of this subgrid re-distribution on the LSM grid is smoothed. Therefore, the effect of lateral terrestrial water flow on simulated precipitation could increase with the increase of the LSM grid's horizontal resolution.

Following these remarks, the original plan was to focus on the European case and to generate an ensemble of 10 WRF and 10 WRF-Hydro simulations for the continental-scale domain displayed in Fig. 1, using two different horizontal grid spacing for the LSM grid, namely 5 km and 2.5 km. The study period was set to June-July-August 2008, as relevant results for the Central Europe region are already available for this period (Arnault et al. [4]), with a spinup period set to 5 months from January to May 2008. For the ensemble generation, the Stochastic Kinetic-Energy

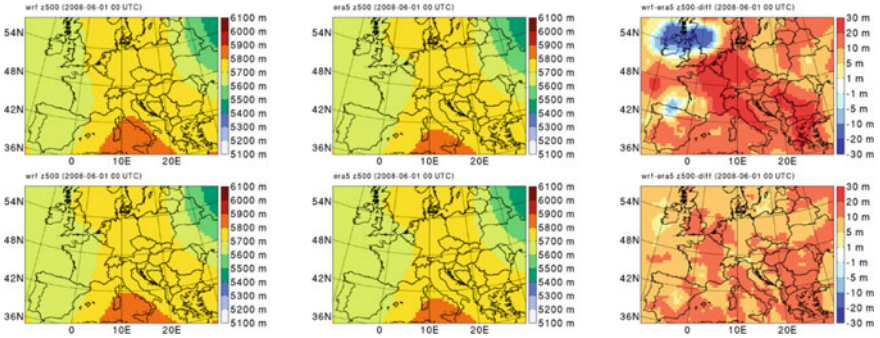


Fig. 1 Upper row: geopotential heights at 500 hPa from a WRF simulation (left) and ERA5 (Copernicus Climate Change Service, 2017) input data (middle) on 1 June 2008 with a model initialization on 1 January 2008. No spectral nudging is considered in this WRF simulation. The right panel shows the difference between WRF and ERA5. Lower row: As the upper row, except for another WRF simulation including spectral nudging.

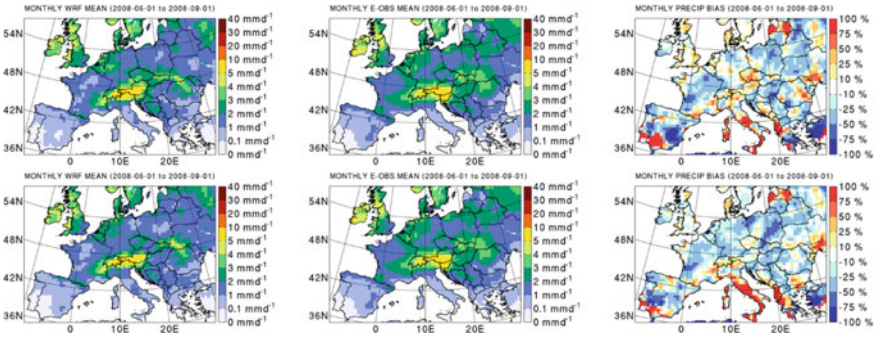


Fig. 2 As in Fig. 1, except for the mean precipitation between 1 June and 1 September 2008, and observational product from E-OBS [10]. Upper row: without spectral nudging; Lower row: with spectral nudging.

Backscatter Scheme (SKEBS, Berner et al. [7]) was selected in order to generate ensemble members with comparable model performance. The surface evaporation tagging and infiltration excess tagging, based on the soil-vegetation-atmospheric water tagging procedure of Arnault et al. [5], were also to be applied in order to quantify the impact of WRF-Hydro-enhanced surface evaporation on continental-scale precipitation. Finally, as the simulation domain is relatively large, it was envisaged to use the spectral nudging method of Miguez-Macho et al. [17] in order to prevent the simulated synoptic circulation deviating from that in the input reanalysis data.

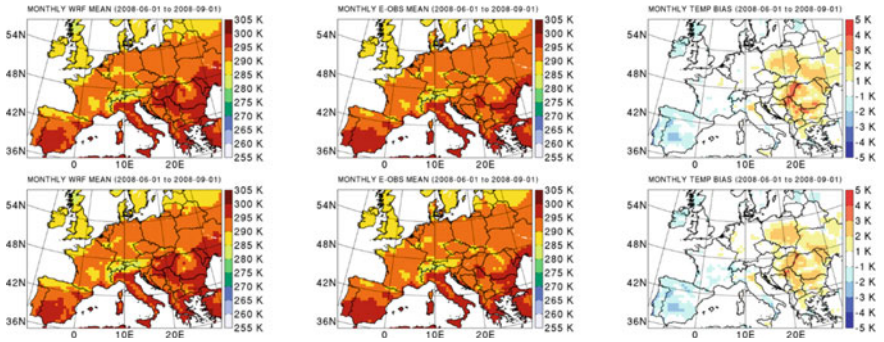


Fig. 3 As in Fig. 1, except for the mean temperature between 1 June and 1 September 2008, and observational product from E-OBS [10]. Upper row: without spectral nudging; Lower row: with spectral nudging.

2.2.2 Tests and Modifications Brought to the Research Plan

Spectral Nudging

The spectral nudging method of Miguez-Macho et al. (2005) has been tested for a WRF simulation with a 5 km grid spacing. The added value of spectral nudging can be clearly seen on geopotential heights at 500 hPa, as illustrated in Fig. 1. Indeed, modelled geopotential heights at 500 hPa are much closer to those in the ERA5 input data when spectral nudging is activated. However, the model result difference in precipitation with and without spectral nudging is small, as can be assessed in Fig. 2. Still, the warm bias in eastern Europe is slightly reduced when spectral nudging is activated, as shown in Fig. 3. Overall, the beneficial impact of using spectral nudging in this case does not seem to be worth the additional computational effort, so that in the following of this project it is chosen to conduct the WRF and WRF-Hydro ensembles without spectral nudging.

Infiltration Excess Tagging with WRF-Hydro

In the original plan it was envisaged to generate two sets of WRF-Hydro ensemble, the first one with the tagging of continental surface evaporation, and the second one with the tagging of infiltration excess. The idea was to quantify the fate of this infiltration excess in the WRF-Hydro simulation and relate it to some precipitation characteristics, which could potential highlight a process-based precipitation difference with a WRF simulation in which the fate of this infiltration excess is neglected. In particular, the hope was that the tagged precipitation originating from infiltration excess in WRF-Hydro would be comparable to the difference in precipitation between WRF and WRF-Hydro. However, the reviewer was not convinced by this argumentation and the allocated computing resources were cut accordingly. Still, we wanted to test this idea. In order to save computing resource, the tagging procedure has been modified in order to allow for two tagged water cycles within one simulation. With this new development, it is now possible to generate a single set of WRF-Hydro ensemble including both surface evaporation tagging and infiltration excess tagging,

which is of course much cheaper than generating two sets of WRF-Hydro ensemble with one tagged water cycle each. However, the result was deceiving. It appeared that the tagged precipitation originating from infiltration excess in WRF-Hydro was much larger than the difference in precipitation between WRF and WRF-Hydro. This is related to the fact that the infiltration excess which re-infiltrates in WRF-Hydro not only increases the amount of soil moisture, but also accelerates the infiltration and percolation below the bottom soil layer. As a consequence, the amount of tagged soil moisture originating from infiltration excess in WRF-Hydro is much larger than the difference in soil moisture between WRF and WRF-Hydro, leading to an amount of tagged precipitation originating from infiltration excess in WRF-Hydro much larger than the difference in precipitation between WRF and WRF-Hydro. So finally, this tagging of infiltration excess was not considered as useful for this particular investigation and therefore abandoned, as suggested by the reviewer.

Effect of Horizontal Resolution—Test with a Grid Spacing of 2.5 km
Concerning the simulations with a 2.5 km grid spacing, the timestep had to be reduced from 15 s to 12 s in order to prevent from numerical instabilities. Still, one member at 2.5 km has been generated in order to compare it with a member at 5 km. Differences shown in Fig. 4 suggest that the simulation at 2.5 km tends to generate more surface evaporation and more precipitation, although the generation of an ensemble mean would be necessary to confirm this. It would of course be interesting to investigate what would be the effect of the lateral terrestrial water flow when the horizontal resolution is enhanced. However, results at 5 km suggest that the number of 10 members may not be enough to fully disentangle the respective effects of atmospheric random fluctuation and lateral terrestrial water flow-related surface evaporation enhancement on precipitation (see result Sect. 2.2.4).

Updated Research Plan
Since the allocated computing resources do not allow for a large increase of the originally envisaged ensemble size at 2.5 km, the research plan has been modified as follows:

- No ensemble generation at 2.5 km
- No spectral nudging
- No infiltration excess tagging for the WRF-Hydro simulations
- Land surface evaporation tagging for the WRF and WRF-Hydro simulations
- Activation of the SKEBS parameterization with a different seed number for each simulation in order to ensure that the ensemble members are all different from each other
- Study period extended to June-September 2008, that is 122 days, instead of June-August 2008 in order to consider all the warm months
- Duplication of the study for West Africa using the same model setup, as one original goal of W2W-A5 was to compare the effect of lateral terrestrial water flow on precipitation in both European and West African regions
- Extension of the ensemble size to 20 WRF members and 20 WRF-Hydro members, eventually more if enough additional computing resource is allocated.

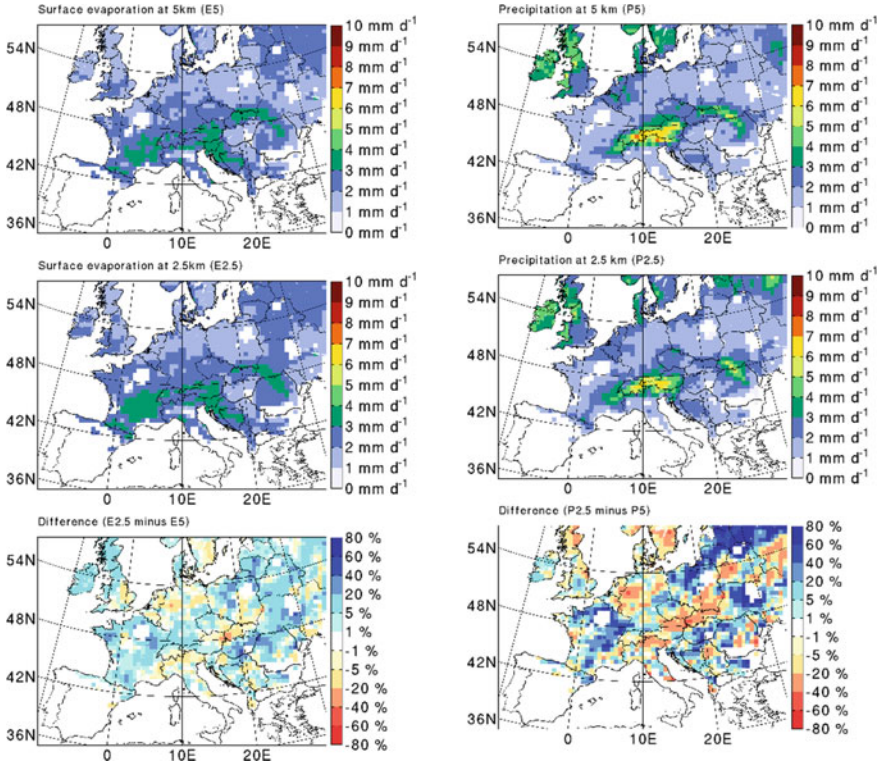


Fig. 4 (First column) mean surface evaporation between 1 June and 1 September 2008 from a WRF simulation with 5 km grid spacing (top), from a WRF simulation with 2.5 km grid spacing (middle) and difference between the two (bottom). (second column) As in the first column, except for precipitation.

2.2.3 Simulation Strategy and Current Status of the Simulations Completed

The performance of the considered model setup at ForHLR2 has been assessed with a series of scaling tests (see Fig. 5). As it is envisaged to generate ensembles, it is chosen to run the simulations with 8 nodes.

The computation of each member is split in 122 consecutive jobs, one job for each day. The job for the first day is submitted to the ForHLR2 super computer. When this job is done, it automatically submits the job for the next day etc. As a result, there is an output file for each day, so 122 output files per member, each output file having of size of about 400 Mb. This gives a total size of about 50 Gb per member.

Using $8 \times 20 = 160$ processors, a single day of simulation (one job) at ForHLR2 with WRF takes about 1 h 45 min, while with WRF-Hydro it takes about 2h40 min. Accordingly, an ensemble with 10 WRF members and 10 WRF-Hydro members costs 863,000 CPUh and can theoretically be achieved in 14 days if there is no queuing time. In practice, we experience that the time to achieve the parallel computation of

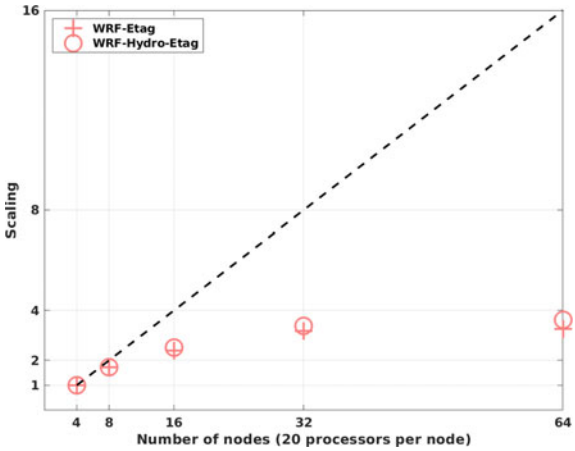


Fig. 5 Scaling of the WRF and WRF-Hydro simulations enhanced with surface evaporation tagging, at ForHLR2

Table 1 Summary of CPUh used at ForHLR2. Miscellaneous includes spinup simulations, simulations at 2.5 km, simulations with infiltration excess tagging, and also simulations with a bugged version of the surface evaporation tagging.

Simulations completed	CPU used
20 WRF members and 20 WRF-Hydro members for Europe	1,726,000
10 WRF members and 10 WRF-Hydro members for West Africa	863,000
Miscellaneous	~4,000,000

a set of 10 WRF members and 10 WRF-Hydro members at ForHLR2 varies between 1 and 2 months.

The amount of CPUh already used at ForHLR2 is summarized in Table 1. It has to be mentioned here that a preliminary set of ensembles for Europe and West Africa has been generated with a bugged version of the surface evaporation tagging. Indeed, adapting the tagging procedure to isotope modelling, as reported in Sect. 2.3.C, we noticed a spurious diverging temporal evolution between total and isotopic precipitating hydrometeors originating from the microphysics scheme. The problem was that the downward velocity of the tagged precipitating hydrometeors was re-evaluated and underestimated, although the downward velocities of the total and tagged precipitating hydrometeors should be the same. In the case of land surface evaporation tagging for the European continent, this downward velocity underestimation led to a precipitation recycling rate of about 19 %, whereas the correct downward velocity led to a precipitation recycling rate of about 20% (not shown). Although this difference is small, about 1%, it was preferred to re-generate the ensembles.

Including the 6,900,000 CPUh of computing resources which was allocated for this year, and the resource remaining from earlier application, we still have about

1,200,000 CPUh available, which will be used to complete 10 additional WRF members and 10 additional WRF-Hydro members for West Africa.

With respect to the size of stored data at ForHLR2, the total amount of the output files for 20 WRF members and 20 WRF-Hydro members for Europe and West Africa represents about 4 Tb. Temporary files during the computation may increase this number by up to +4 Tb. All this data is currently stored in the workspace directory.

2.2.4 Current Results

Convergence Criteria

The aim of generating WRF and WRF-Hydro ensemble simulations for Europe and West Africa is to disentangle the respective effects of atmospheric random fluctuation and lateral terrestrial water flow-related surface evaporation enhancement on precipitation. Indeed, it is clear that the effect of atmospheric random fluctuation on the precipitation ensemble mean becomes smaller when the ensemble size is increased. The question is to know what is the minimal ensemble size required that the difference between WRF and WRF-Hydro precipitation ensemble mean would be mainly due to the lateral terrestrial water flow-related surface evaporation enhancement. In order to address this specific issue, we have considered the following convergence criteria $c^{nrmsd}(n)$ and $c^{nmd}(n)$, where n is the number of ensemble members:

$$c^{\mathrm{nrmsd}}(n) = \left(\frac{\sum_{(i,j)\in \mathrm{land}} \left(\frac{1}{n+1} \sum_{k=1}^{n+1} \left(P_{\mathrm{WH}}^{i,j,k} - P_{\mathrm{WS}}^{i,j,k} \right) - \frac{1}{n} \sum_{k=1}^{n} \left(P_{\mathrm{WH}}^{i,j,k} - P_{\mathrm{WS}}^{i,j,k} \right) \right)^2}{\sum_{(i,j)\in \mathrm{land}} \left(\frac{1}{n} \sum_{k=1}^{n} \left(P_{\mathrm{WH}}^{i,j,k} - P_{\mathrm{WS}}^{i,j,k} \right) \right)^2} \right)^{0.5} \tag{1}$$

$$c^{\mathrm{nmd}}(n) = \frac{\sum_{(i,j)\in \mathrm{land}} \left(\frac{1}{n+1} \sum_{k=1}^{n+1} \left(P_{\mathrm{WH}}^{i,j,k} - P_{\mathrm{WS}}^{i,j,k} \right) - \frac{1}{n} \sum_{k=1}^{n} \left(P_{\mathrm{WH}}^{i,j,k} - P_{\mathrm{WS}}^{i,j,k} \right) \right)}{\sum_{(i,j)\in \mathrm{land}} \left(\frac{1}{n} \sum_{k=1}^{n} \left(P_{\mathrm{WH}}^{i,j,k} - P_{\mathrm{WS}}^{i,j,k} \right) \right)} \tag{2}$$

In (1) and (2) P_{WS} and P_{WH} stand for the study period-average precipitation from a WRF "standard" simulation and from a WRF-Hydro simulation, respectively. Subscripts i and j refer to the horizontal coordinates on the WRF grid, while subscript k refers to the member number.

$c^{nrmsd}(n)$ is the normalized root mean squared difference between two ensemble mean differences between WRF and WRF-Hydro ensembles, one with an ensemble size of n and the other with an ensemble size of $n + 1$. As such, c^{nrmsd} is a measure of the dependency of the spatial differences between WRF and WRF-hydro ensemble means to the ensemble size n. For sufficiently large n, the effect of atmospheric random fluctuations on these spatial differences is expected to become negligible and $c^{nrmsd}(n)$ to converge towards zero.

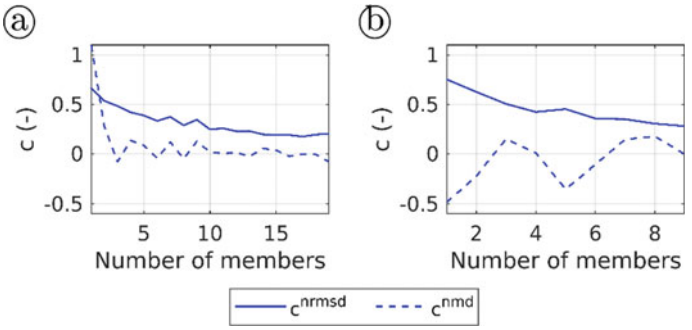


Fig. 6 Timeseries of c^{nrmsd} and c^{nmd} as a function of the number of members, for the European ensemble **a** and for the West African ensemble **b**. Note that the current European ensemble has 20 members, while the West African ensemble only has 10 members.

$c^{nmd}(n)$ is the normalized mean difference between two ensemble mean differences between WRF and WRF-Hydro ensembles, one with an ensemble size of n and the other with an ensemble size of $n + 1$. Therefore, c^{nmd} is a measure of the dependency of the spatially-average difference between WRF and WRF-Hydro ensemble means to the ensemble size. For sufficiently large n, the effect of atmospheric random fluctuations on this spatially-average difference is also expected to become negligible and $c^{nmd}(n)$ to converge towards zero. These two convergence criteria are displayed in Fig. 6 for the European and West African cases.

Figure 6 reveals two important features. Firstly, the spatial differences between the WRF and WRF-hydro ensemble means are still affected by atmospheric random fluctuations for an ensemble size of 20, with a convergence criterion c^{nrmsd} reaching 0.20 for the European case. Secondly, the convergence criterion c^{nmd} appears to converge relatively quickly towards zero when the ensemble size is increased, both for the European and West African cases. This result implies that, for the current size of both the European and West African ensembles, the spatial differences between the WRF and WRF-Hydro ensemble means are partially influenced by atmospheric random fluctuations, whereas the spatially-average difference between the WRF and WRF-Hydro ensemble means can be mainly attributed to lateral terrestrial water flow-related surface evaporation enhancement.

Spatial Analysis

In order to visualize these spatial differences between WRF and WRF-Hydro ensemble means, we consider the normalized ensemble mean difference of surface evaporation E and precipitation P defined as follows:

$$\Delta E^{i,j} = 100 \frac{\sum_{k=1}^{N} \left(E_{\mathrm{WH}}^{i,j,k} - E_{\mathrm{WS}}^{i,j,k} \right)}{0.5 \sum_{k=1}^{N} \left(P_{\mathrm{WH}}^{i,j,k} + P_{\mathrm{WS}}^{i,j,k} \right)} \tag{3}$$

$$\Delta P^{i,j} = 100 \frac{\sum_{k=1}^{N} \left(P_{\mathrm{WH}}^{i,j,k} - P_{\mathrm{WS}}^{i,j,k} \right)}{0.5 \sum_{k=1}^{N} \left(P_{\mathrm{WH}}^{i,j,k} + P_{\mathrm{WS}}^{i,j,k} \right)} \tag{4}$$

N is the ensemble size, currently equal to 20 for the European ensemble and equal to 10 for the West African ensemble. It is recalled that these WRF and WRF-Hydro simulations are enhanced with the tagging of land surface evaporation, so that the variable P^{tag} in the model outputs gives the amount of precipitation originating from evaporated water over the land. Therefore, the difference in P^{tag} between the WRF and WRF-Hydro simulations is directly linked to the lateral terrestrial water flow-related surface evaporation enhancement. The normalized ensemble mean difference of tagged precipitation, which is called P_{land}, is defined as:

$$\Delta P_{\mathrm{land}}^{i,j} = 100 \frac{\sum_{k=1}^{N} \left(P_{\mathrm{WH}}^{tag,i,j,k} - P_{\mathrm{WS}}^{tag,i,j,k} \right)}{0.5 \sum_{k=1}^{N} \left(P_{\mathrm{WH}}^{i,j,k} + P_{\mathrm{WS}}^{i,j,k} \right)} \tag{5}$$

P_{land} quantifies the part of P which originates from the land surface change triggered by the lateral terrestrial water flow. Additionally, the difference between P and P_{land} is interpreted as the effect of atmospheric changes triggered by the lateral terrestrial water flow, and is named P_{atm}:

$$\Delta P_{\mathrm{atm}} = \Delta P - \Delta P_{\mathrm{land}} \tag{6}$$

Spatial patterns of both P_{land} and P_{atm} are also influenced by the atmospheric random fluctuations, at least with the current ensemble size available as deduced by c^{nrmsd} in Fig. 6. Maps of E, P, P_{land} and P_{atm} are displayed in Fig. 7 for the European and West African cases.

E mainly displays positive patterns, thus confirming that the description of lateral terrestrial water flow in WRF-Hydro generally increases the surface evaporation [2, 4, 23]. Some areas where E is negative are related to mountainous regions, where the lateral terrestrial water flow reduces the residence time of the soil water storage in the root zone, and therefore reduces the amount of surface evaporation (e.g. Zhang et al. [29], Arnault et al. [5]). Other areas with negative values of E are more related to precipitation differences shown in P.

P and P_{land} mainly display positive patterns, which corresponds to the lateral terrestrial water flow-related surface evaporation enhancement on precipitation. Interestingly, P_{land} is more uniformly positive in comparison to P, and most of the spatial patterns of P are provided by P_{atm}. This suggests that P_{atm}, which is the atmospheric change-related precipitation difference triggered by the lateral terrestrial water flow, is significantly affected by atmospheric random fluctuations. Still, the fact that c^{nrmsd} in Fig. 6 is relatively small when the ensemble size is set to 20 suggests that P_{atm} is not a pure random process.

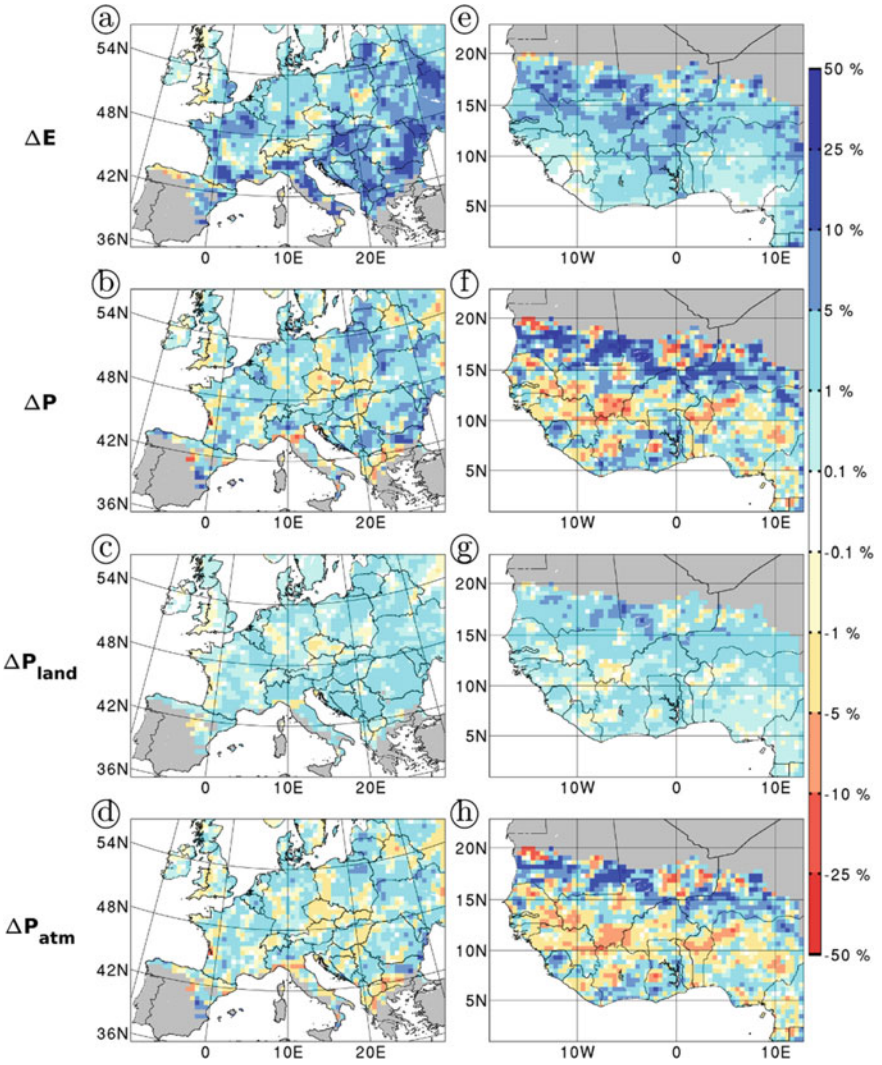


Fig. 7 Maps of E, P, P_{land} and P_{atm} for Europe (left column) and West Africa (right column). Note that the current European ensemble has 20 members, while the West African ensemble only has 10 members.

Concluding Remarks

In the case of Europe, the above result implies that the lateral terrestrial water flow tends to increase precipitation by two ways. Firstly, it generally increases soil storage, surface evaporation and precipitation through recycling. Secondly, the increased surface evaporation tends to trigger more cloud condensation and therefore to convert more atmospheric water vapor into cloud water and precipitation. However, this

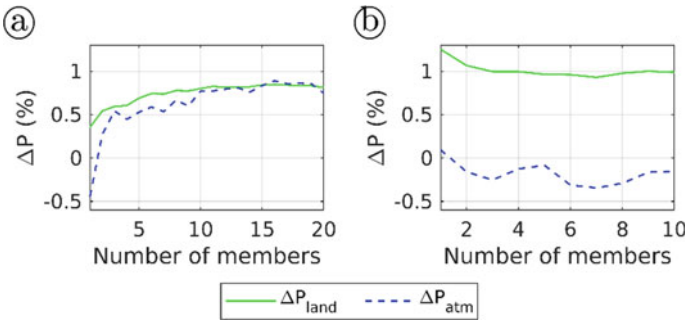


Fig. 8 Timeseries of P_{land} and P_{atm} as a function of number of members for the ensemble size, for the European ensemble **a** and for the West African ensemble **b**. Note that the current European ensemble has 20 members, while the West African ensemble only has 10 members.

second process is highly dependent on atmospheric random fluctuations which may have a larger impact on the amount of cloud condensation occurring at a specific place. The contribution of these two processes to the lateral terrestrial water flow-related enhancement of precipitation is well summarized with spatially-average P_{land} and P_{atm}, displayed as a function of the ensemble size in Fig. 8.

In the case of West Africa, the effect of lateral terrestrial water flow on P_{atm} seems to be latitude-dependant, with more positive values in the northern part of the Sahelian region and with more negative values in the southern part of the Sahelian region. A negative P_{atm} could be related to the fact that the lateral terrestrial water flow-related surface evaporation enhancement does not increase cloud condensation, but instead the evaporated water from the land takes the place of the atmospheric water vapour, so that the contribution of atmospheric water vapour to precipitation in a WRF-Hydro simulation is less than in a WRF simulation.

It has to be acknowledged that these results are preliminary as they are to some extend affected by random atmospheric fluctuation. A larger ensemble size is required here in order to reduce this random effect and improve the quality of the results.

2.2.5 Next Steps

As already mentioned above at Sect. 2.2.3, the ensemble size for West Africa first has to be increased to 20, in order to allow for a fair comparison between the European and West African results. In the future it would be valuable to increase the size of the ensemble to 40 members in order to reduce the contribution of atmospheric random fluctuations and better quantify the contribution of lateral terrestrial water flow to precipitation. This additional computational effort will be part of a follow-up proposal.

Differences between European and West African ensemble results still need to be analysed in more details. This analysis will also include the evaluation of the effect of lateral terrestrial water flow on model internal variability, by adapting the method developed by Arnault et al. [4] for the Central European region. In the end, these

results will provide the material for a paper focusing on the lateral terrestrial water flow feedbacks to summer precipitation at continental scale (Arnault et al. [6], in revision).

References

1. R.O. Anyah, C.P. Weaver, G. Miguez-Macho, Y. Fan, A. Robock, Incorporating water table dynamics in climate modeling: 3. Simulated groundwater influence on coupled land-atmosphere variability. J. Geophys. Res. **113**, D0703 (2008)
2. J. Arnault, S. Wagner, T. Rummler, B. Fersch, J. Bliefernicht, S. Andresen, H. Kunstmann, Role of runoff-infiltration partitioning and resolved overland flow on land-atmosphere feedbacks: a case study with the WRF-Hydro coupled modeling system for West Africa. J. Hydrometeor. **17**, 1489–1516 (2016a)
3. J. Arnault, R. Knoche, J. Wei, H. Kunstmann, Evaporation tagging and atmospheric water budget analysis with WRF: a regional precipitation recycling study for West Africa. Water Resour. Res. **52**, 1544–1567 (2016b)
4. J. Arnault, T. Rummler, F. Baur, S. Lerch, S. Wagner, B. Fersch, Z. Zhang, N. Kerandi, C. Keil, H. Kunstmann, Precipitation sensitivity to the uncertainty of terrestrial water flow in WRF-Hydro: an ensemble analysis for Central Europe. J. Hydrometeor. **19**, 1007–1025 (2018)
5. J. Arnault, J. Wei, T. Rummler, B. Fersch, Z. Zhang, G. Jung, S. Wagner, H. Kunstmann, A joint soil-vegetation-atmospheric water tagging procedure with WRF-Hydro: implementation and application to the case of precipitation partitioning in the upper Danube river basin. Water Resour. Res. **55**, 6217–6243 (2019)
6. J. Arnault, B. Fersch, T. Rummler, Z. Zhang, G.M. Quenum, J. Wei, M. Graf, P. Laux, H. Kunstmann, Lateral terrestrial water flow contribution to summer precipitation at continental scale—a comparison between Europe and West Africa with WRF-hydro-tag ensembles. Process. Hydrol. (2020). (in revision)
7. J. Berner, G.J. Shutts, M. Leutbecher, T.N. Palmer, A spectral stochastic kinetic energy backscatter scheme and its impact on flow-dependent predictability in the ECMWF ensemble prediction system. J. Atmos. Sci. **66**, 603–626 (2009)
8. Copernicus Climate Change Service (C3S), ERA5: fifth generation of ECMWF atmospheric reanalyses of the global climate. Copernicus Climate Change Service Climate Data Store (CDS) (2017). https://cds.climate.copernicus.eu/cdsapp#!/home
9. D.J. Gochis, W. Yu, D.N. Yates, The WRF-hydro model technical description and user's guide, version 3.0, NCAR Technical Document, 120 p (2015). http://www.ral.ucar.edu/projects/wrf_hydro/
10. M.R. Haylock, N. Hofstra, A.M.G. Klein Tank, E.J. Klok, P.D. Jones, M. New, A European daily high-resolution gridded dataset of surface temperature and precipitation. J. Geophys. Res. **113**, D20119 (2008)
11. S.-Y. Hong, J.-O.J. Lim, The WRF single-moment 6-class microphysics scheme (WSM6). J. Korean Meteor. Soc. **42**, 129–151 (2006)
12. G.J. Huffman, R.F. Adler, D.T. Bolvin, G. Gu, E.J. Nelkin, K.P. Bowman, Y. Hong, E.F. Stocker, D.B. Wolff, The TRMM multi-satellite precipitation analysis: quasi-global, multi-year, combined-sensor precipitation estimates at fine scale. J. Hydrometeor. **8**, 38–55 (2007)
13. N. Kerandi, J. Arnault, P. Laux, S. Wagner, J. Kitheka, H. Kunstmann, Joint atmospheric-terrestrial water balances for East Africa: a WRF-hydro case study for the upper Tana River basin. Theor. Appl. Climatol. **131**, 1337–1355 (2017)
14. M.A.D. Larsen, J.H. Christensen, M. Drews, M.B. Butts, J.C. Refsgaard, Local control on precipitation in a fully coupled climate-hydrology model. Sci. Rep. **6**, 22927 (2016)

15. R.M. Maxwell, F.K. Chow, S.J. Kollet, The groundwater-land-surface-atmosphere connection: soil moisture effects on the atmospheric boundary layer in fully-coupled simulations. Adv. Water Resour. **30**, 2447–2466 (2007)
16. R.M. Maxwell, J.K. Lundquist, J.D. Mirocha, S.G. Smith, C.S. Woodward, A.F. Tompson, Development of a coupled groundwater-atmosphere model. Mon. Weather Rev. **139**, 96–116 (2011)
17. G. Miguez-Macho, G.L. Stenchikov, A. Robock, Spectral nudging to eliminate the effects of domain position and geometry in regional climate model simulations. J. Geophys. Res. **109**, D13104 (2004)
18. G.-Y. Niu, Z.-L. Yang, K.E. Mitchell, F. Chen, M.B. Ek, M. Barlage, A. Kumar, K. Manning, D. Niyogi, E. Rosero, M. Tewari, Y. Xia, The community Noah land surface model with multiparameterization options (Noah-MP): 1. Model description and evaluation with local-scale measurements. J. Geophys. Res. **116**, 11 (2011)
19. J. Nusbaumer, T.E. Wong, C. Bardeen, D. Noone, Evaluating hydrological processes in the community atmosphere model version 5 (CAM5) using stable isotope ratios of water. J. Adv. Model. Earth Syst. **9**, 949–977 (2017)
20. S. Pfahl, H. Wernli, K. Yoshimura, The isotopic composition of precipitation from a winter storm—a case study with the limited-area model COSMOiso. Atmos. Chem. Phys. **12**, 1629–1648 (2012)
21. J.E. Pleim, A combined local and nonlocal closure model for the atmospheric boundary layer. Part I: model description and testing. J. Appl. Meteor. Climatol. **46**, 1383–1395 (2007)
22. M. Rahman, M. Sulis, S.J. Kollet, The subsurface-land surface-atmosphere connection under convective conditions. Adv. Water Resour. **83**, 240–249 (2015)
23. T. Rummler, J. Arnault, D. Gochis, H. Kunstmann, Role of lateral terrestrial water flow on the regional water cycle in a complex terrain region: investigation with a fully coupled model system. J. Geophys. Res. Atmos. **124**, 507–529 (2019)
24. A. Senatore, G. Mendicino, D.J. Gochis, W. Yu, D.N. Yates, H. Kunstmann, Fully coupled atmosphere-hydrology simulations for the central Mediterranean: impact of enhanced hydrological parameterization for short and long time scales. J. Adv. Model. Earth Syst. **7**, 1693–1715 (2015)
25. W.C. Skamarock, J.B. Klemp, A time-split nonhydrostatic atmospheric model for weather research and forecasting applications. J. Comp. Phys. **227**, 3465–3485 (2008)
26. K.E. Trenberth, Atmospheric moisture recycling: role of advection and local evaporation. J. Clim. **12**, 1368–1381 (1999)
27. S. Wagner, B. Fersch, F. Yuan, Z. Yu, H. Kunstmann, Fully coupled atmospheric-hydrological modeling at regional and long-term scales: development, application, and analysis of WRF-HMS. Water Resour. Res. **52**, 3187–3211 (2016)
28. T.E. Wong, J. Nusbaumer, D.C. Noone, Evaluation of modeled land-atmosphere exchanges with a comprehensive water isotope fractionation scheme in version 4 of the community land model. J. Adv. Model. Earth Syst. **9**, 978–1001 (2017)
29. Z. Zhang, J. Arnault, S. Wagner, P. Laux, H. Kunstmann, Impact of lateral terrestrial water flow on land-atmosphere interactions in the Heihe River Basin in China: fully coupled modeling and precipitation recycling analysis. J. Geophys. Res. Atmos. **124**, 8401–8423 (2019)

Computer Science

Hans-Joachim Bungartz

The following part deals with research labelled as "Computer Science", mostly due to the fact that the respective groups are affiliated with Computer Science departments. However, if we look at the domains of application addressed in the five annual reports submitted this year, the impression is a bit different: we find molecular dynamics, coupled problems, and others. Nevertheless, it is right that the focus of the underlying research is on typical informatics issues, such as load balancing, fault tolerance, GPU acceleration, or scalability – which justifies the classification for this volume.

Out of those five submissions undergoing the usual reviewing process, two project reports were selected: GCS-MDDC and lamd.

The contribution *Load Balanced Particle Simulation with Automated Algorithm Selection* by Neumann et al. reports on recent progress in the GCS-MDDC project. The project's basis are, first, the software *ls1 mardyn*, a molecular dynamics framework for multi-phase and multi-component studies at small scales with applications in process engineering, and, second, the library *AutoPas* that employs auto-tuning across a wide range of particle simulation schemes, data structures, and node-level parallelization patterns. Both tools have been developed by the groups of the authors and collaborators. The focus of the research reported in this article is on a sophisticated integration of *AutoPas* into *ls1 mardyn*, as well as the extension of the load balancing portfolio by diffusive load balancing.

The second report, *Load-Balancing for Large-Scale Soot Particle Agglomeration Simulations* by Hirschmann et al., deals with the project lamd. Being similar to the first report in terms of the underlying simulation methodology, lamd uses *ESPResSo* as its software basis. The focus of the research activities in the last year was on combining previous efforts to simulate large-scale soot particle agglomerations with a dynamic

H.-J. Bungartz
Department of Informatics, Technische Universität München (TUM)
Boltzmannstrasse 3, 85748 Garching, Germany
e-mail: bungartz@in.tum.de

turbulent backgound flow field, which resulted in a significantly increased number of particles that can be tackled.

By chance, both papers use short-range molecular dynamics and sophisticated load balancing, but they also differ in various respects. Together, they show the importance of algorithmic improvements for advancing the frontiers of supercomputing.

Load Balanced Particle Simulation with Automated Algorithm Selection

Philipp Neumann, Fabio Gratl, Steffen Seckler, and Hans-Joachim Bungartz

Abstract *ls1 mardyn* is a molecular dynamics (MD) simulation framework for multiphase and multicomponent investigations at small scales with application in process engineering. *AutoPas* is a library that employs auto-tuning for various particle simulation algorithms, data structures, and node-level parallelization patterns.

In the last year, we have focussed on improving the interfaces of *AutoPas* to support massively parallel, distributed-memory simulations. In this scope, we have revised the incorporation of *AutoPas* into *ls1 mardyn* and extended *ls1 mardyn* by diffusive load balancing.

1 Introduction

Molecular dynamics (MD) simulations have become a valuable tool for engineering applications. Based on Newton's laws of motion

$$\frac{d^2\mathbf{x}_p}{dt^2} = \frac{1}{m_p}\mathbf{F}_p,$$

P. Neumann (✉)
Helmut-Schmidt-Universität Hamburg, Department of Mechanical Engineering,
Holstenhofweg 85, 22043 Hamburg, Germany
e-mail: philipp.neumann@hsu-hh.de

F. Gratl · S. Seckler · H.-J. Bungartz
Technical University of Munich, Department of Informatics,
Boltzmannstr. 3, 85748 Garching, Germany
e-mail: gratl@in.tum.de

S. Seckler
e-mail: seckler@in.tum.de

H.-J. Bungartz
e-mail: bungartz@in.tum.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_33

the positions $\mathbf{x}_p$ of individual particles p can be tracked virtually. These particles move due to forces $\mathbf{F}_p = \sum_{q \neq p} \mathbf{F}_{pq}$, arising from interacting forces between particles p, q. We will concentrate on short-range interactions: only particles within a prescribed distance—the so-called *cut-off radius* r_c—interact with each other. This can be efficiently realized algorithmically by sorting the particles into Cartesian grid cells (linked cell method) that exhibit the mesh size r_c, or by storing potential interaction partners for each particle in lists (Verlet lists) [9]; to avoid the expensive construction of particle pair lists, these lists are only built every few time steps and comprise all particle pairs within a distance $r_c + s$ where s is the so-called skin radius. If s is rather large, the lists have to be built less frequently, but this comes at increasing particle pair traversal cost.

Amongst others, MD simulations can be used to sample thermodynamic properties from large systems of small molecules. This allows, e.g., to determine equations of state for complex fluids, to investigate bubble formation [5], interfacial flows [6] or droplet coalescence [10].

In a long-standing, interdisciplinary collaboration of computer scientists and mechanical engineers, the MD simulation software *ls1 mardyn* has evolved to assess such systems [7, 15]. *ls1 mardyn* supports vectorization, shared- as well as distributed-memory parallelism [2, 14, 16] including dynamic load balancing which allows accounting for load imbalances due to heterogeneous particle distributions or due to the use of heterogeneous hardware [11, 12].

Besides load balancing on distributed-memory systems, another grand challenge is given by the fact that a great variety of algorithmic realizations are available to simulate short-range MD systems [1, 4, 8, 9, 17]. An optimal algorithm/data structure layout/parallelization approach (at shared-memory level) strongly depends on both the underlying hardware and the actual problem to solve. With new hardware rapidly evolving and with the heterogeneity of hardware design still increasing in the HPC sector, finding and incorporating the best algorithm into a simulation software is thus demanding.

Over the last years, the authors have developed the particle simulation library *AutoPas* [3], which provides various data structures, particle traversal schemes, and shared-memory parallelization approaches and which automatically selects the optimal combination thereof at run time.

In the following, we detail recent developments that build upon and extend the load balancing capabilities of *ls1 mardyn* as well as our prototypical integration of *AutoPas* and *ls1 mardyn* from the previous report [13]. We discuss the integration of an alternative load balancing library, cf. Sect. 2, and elaborate on challenges when integrating *AutoPas* and *ls1 mardyn* for distributed-memory simulations, cf. Sect. 3. We close with a summary and an outlook to future work in Sect. 4.

Parts of this work have been submitted for publication [11].

2 Load Balancing

So far, *ls1 mardyn* employed a kd-tree-based decomposition to balance the load between MPI processes. Its recursive bisectioning is an efficient means to partition and distribute loads. Yet, the implemented method bears some disadvantages. Due to successive bisectioning, every dimension is visited a multiple number of times in the partitioning process. Besides, a global collection of computational loads, that is particle or at least cell-averaged data, is required and re-balancing can potentially alter the overall topology of the simulation partitioning significantly from one to the next time step.

Due to these points, the *A Load-balancing Library (ALL)*[1] which is developed at the Jülich Supercomputing Center has been incorporated as an alternative to the existing kd-tree implementation. Still following the hierarchical/recursive splitting approach, *ALL* allows to split the domain into multiple subsections in one recursion step and limits the number of splits along every dimension to one. Moreover, to avoid rigorous repartitioning, diffusive load balancing was incorporated into *ls1 mardyn*. Since the diffusive load balancing algorithm operates (more or less) strictly locally, loads are migrated between neighboring processes only. This limits the significance of changes in the parallel topology.

The diffusive load balancing in conjunction with the *ALL*-based partitioner was evaluated in various scenarios, including

- scenario *droplet coalescence*: the coalescence of two nanodroplets that are suspended in a vapor phase, cf. Fig. 1(a),
- scenario *spinodal decomposition*: a fluid is rapidly cooled down and separates into vapor and liquid phases, cf. Fig. 1(b),
- scenario *exploding liquid*: a compressed, hot liquid expands in vacuum, cf. Fig. 1(c).

While no performance improvements were observed in the *spinodal decomposition*—which was expected, since the occurring inhomogeneities are rather fine-scale—, both kd-tree- and diffusive *ALL*-based load balancing significantly boosted performance in the *droplet coalescence*. This is in accordance with expectations since the merging process of the two droplets is relatively slow and thus, load changes occur on rather long time scales. In contrast, diffusive *ALL*-based load balancing outperformed the kd-tree approach significantly for the *exploding liquid*: here, the overall particle distribution in the entire computational domain changes drastically, with particles getting sucked outward at the beginning and bouncing back from the outer boundaries afterwards. This rather directed motion of the film fragments is well reflected by the diffusive load balancing algorithm, which effectively propagates loads between spatially neighbored subdomains and MPI processes, respectively.

[1] https://e-cam.readthedocs.io/en/latest/Meso-Multi-Scale-Modelling-Modules/modules/ALL_library/tensor_method/readme.html.

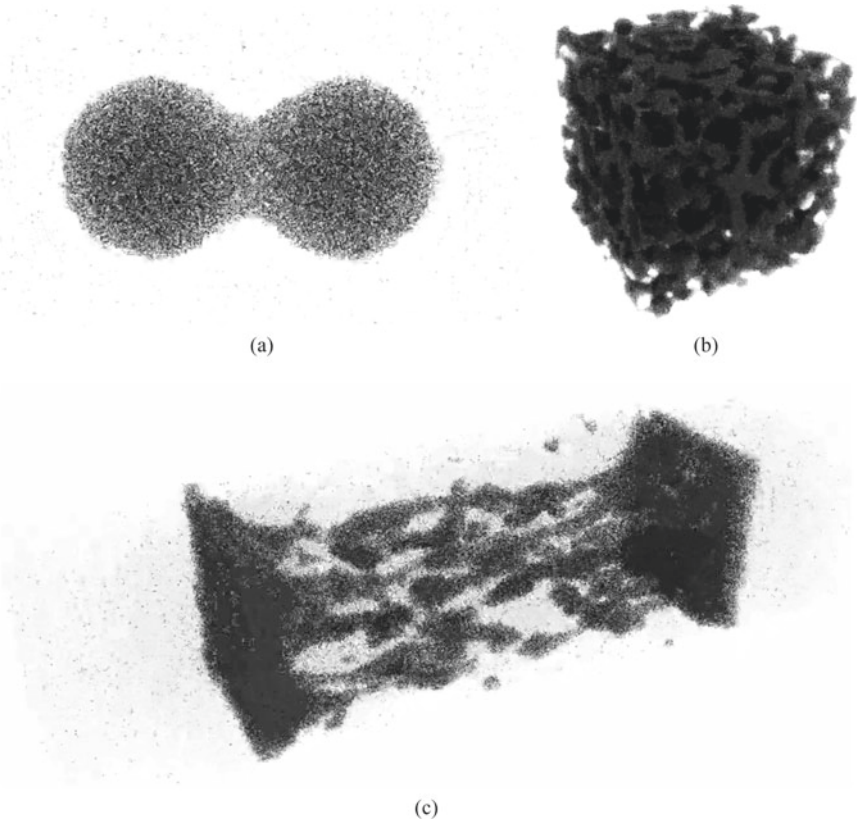
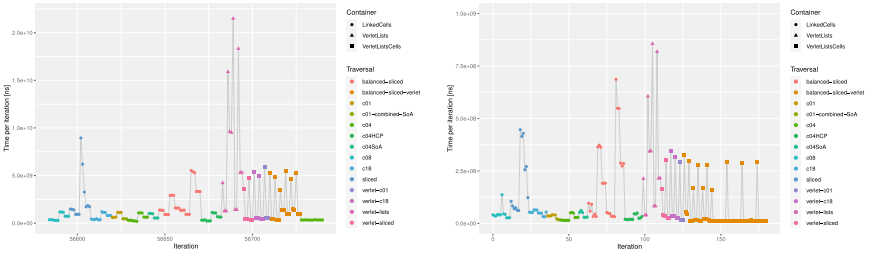


Fig. 1 Scenarios considered in our work. (a) Droplet coalescence. (b) Spinodal decomposition. (c) Exploding liquid; the liquid is initially placed in a vertical box-like form in the middle of the domain and expands/breaks up towards the left and the right of the computational domain

3 Integrating AutoPas and Ls1 Mardyn for Distributed-Memory Simulations

In a prior report [13], we had commented on the integration of *AutoPas* into *ls1 mardyn*, enabling an "on-the-fly" exchange of data structures, particle pair traversal schemes, and parallelization methods during an MD simulation. *AutoPas* already comprised a few implementations and configuration possibilities in terms of data structures, linked cell-based particle traversal schemes, and linked cell-based shared-memory parallelization approaches. At this stage, it had already been possible to execute the *spinodal decomposition* in node-level simulations and to automatically tune over some configurations.

Over the last year, *AutoPas* was significantly extended. In particular, Verlet lists as an alternative to the linked cell method to traverse particle pairs for force interactions



(a) *Exploding liquid*, mid of simulation, inhomogeneous

(b) *Spinodal decomposition*, start of simulation, homogeneous

Fig. 2 Snapshot of simulation performance during the auto-tuning phase. In this phase, 48 algorithmic configurations are tested, employing an exhaustive search. Data points are colored by their traversal choice and the symbol is used to identify the particle container; note that further variants (AoS vs SoA data structures; Newton3-optimization to reduce computations; load balancing schemes) are partly employed and varied (resembling multiple occurrences of the same symbols in the plot). Always exactly three data points represent the same configuration, corresponding to three samples being taken. The x-axis shows the number of iterations, the y-axis shows the time for a single iteration. (a) *Exploding liquid*; the configuration, that is tested last, is chosen and is taken over for the following 5000 iterations. (b) *Spinodal decomposition*.

were incorporated, including a vectorizing variant thereof (cluster lists, as used in the software Gromacs, for example). This yields a total of > 70 discrete algorithmic configurations, cf. [11] for details.

In Fig. 2, the performance behavior of the exhaustive search tuning strategy is visualized. Every available algorithmic configuration is measured over three iterations each. Afterwards, the fastest one is selected for the next period of—in this case—5000 iterations. Verlet lists appear to be fluctuating in performance. This is, however, due to the fact that in their first iteration, the neighbor lists are built, which is taken into account here. As the two setups differ in their structure, differences in the performance profiles can be seen, especially between the faster configurations. It also becomes apparent, that there is still work to be done in a) improving the average performance of individual configurations and b) only testing promising configurations. Both of these tasks are subject to current research.

A specific challenge, however, arises when distributed-memory, that is MPI-parallel, MD simulations shall be supported by AutoPas. Due to potential particle distribution inhomogeneities over all MPI processes, cf. Sect. 2, every MPI process might require a different *AutoPas* configuration to execute at optimal performance, cf. Fig. 3. In particular, one MPI process might make use of Verlet lists, which requires less MPI-based particle data exchange with neighboring processes (i.e., some information only needs to be exchanged every few time steps), while another MPI process might employ linked cells which require particle data exchange in every time step. This also renders the handling of particles at the interface of the embedding simulation (*ls1 mardyn* in our case) and *AutoPas* challenging, since particle exchange and updates need to be carried out consistently.

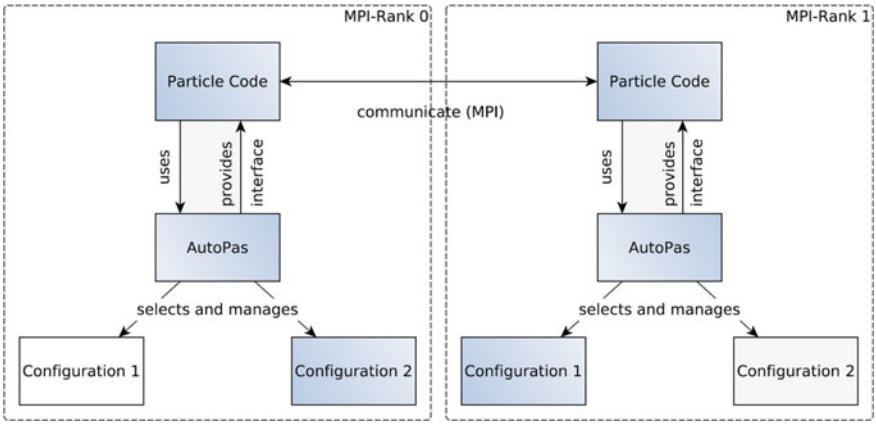


Fig. 3 MPI-parallel use of *AutoPas* in *ls1 mardyn*. *AutoPas* may execute different configurations on different MPI processes. Yet, a common interface and semantics is required to bridge all potential particle simulation configurations of *AutoPas*

To solve this, the interface to *AutoPas* was designed such that it follows the Verlet list idea, in the sense that the particle container of *AutoPas* is only being updated every few time steps. This implies that the linked cell implementation of *AutoPas* needs to operate on slightly enlarged cells, corresponding to the skin radius s that is typically added to the cut-off radius r_c in the Verlet list approach to avoid frequent list rebuilds. Yet, refined linked cells are also supported, which means that more neighbor cells need to be searched. Although the Verlet-like approach comes with rather good code maintainability features and performance, the overall usability of the library, especially the particle exchange in the *AutoPas*-embedding MD simulation, becomes slightly more advanced.

Putting things together, we have combined *ALL*-based diffusive load balancing and *AutoPas*-based automated node-level algorithm selection. Figure 4 shows the different algorithm choices per subdomain of an exemplary load balanced domain decomposition for the scenario *droplet coalescence*. We simulated the three aforementioned scenarios *droplet coalescence*, *exploding liquid* and *spinodal decomposition* with this simulation technology on two platforms. Due to the HLRS machine HAWK getting set up in 2019, we first established and tuned simulations on SuperMUC-NG at LRZ during that time. Figure 5 shows the scalability for the *exploding liquid* on both machines. We can first observe that, expectedly, the per-node performance on HAWK is significantly better than on SuperMUC-NG in terms of simulated time steps per second. The *ALL*-based load balancing with auto-tuning via *AutoPas* exhibits best performance on SuperMUC-NG compared to the other configurations. This approach also scales well up to 8 nodes on HAWK. For larger node counts, however, the curve currently flattens. Reasons for this lie in one node featuring 8 NUMA domains; since we use one MPI process per NUMA domain, this yields significantly more MPI processes on HAWK compared to SuperMUC-NG.

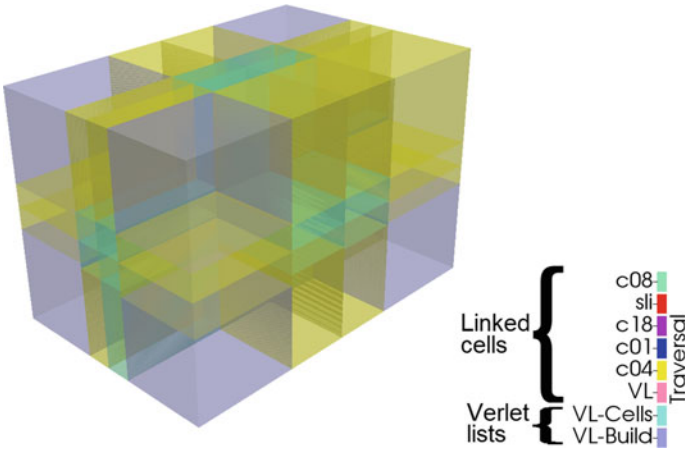


Fig. 4 Algorithm selection on a load-balanced simulation of *droplet coalescence*. c08, sli, c18, c01, c04 correspond to OpenMP parallelization schemes that operate on linked cells. In contrast, VL, VL-Cells, VL-Build are parallel schemes that operate on Verlet lists

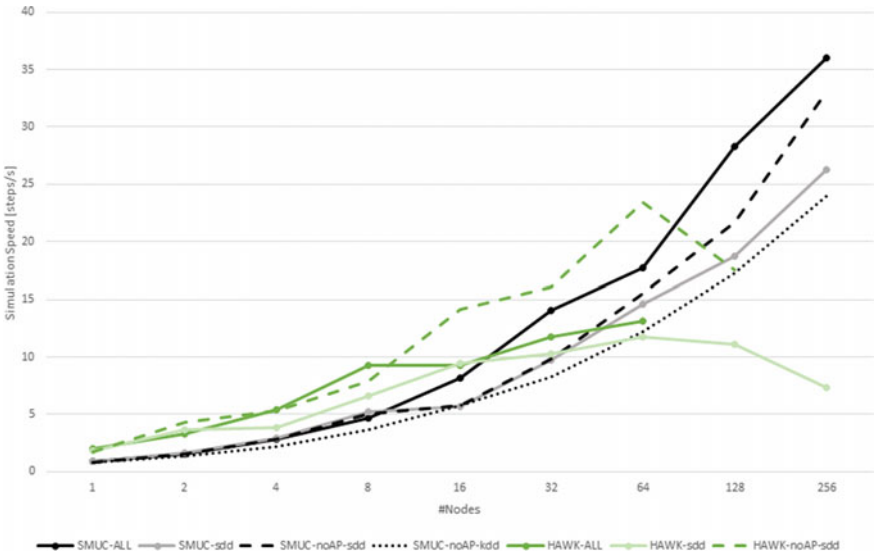


Fig. 5 Scalability of the scenario *exploding liquid* on two platforms SuperMUC-NG and HAWK. –ALL: *ALL*-based diffusive load balancing; –sdd: standard domain decomposition; –noAP-sdd: standard domain decomposition with *ls1 mardyn* kernels instead of using *AutoPas*; –noAP-kdd: kd-tree decomposition with *ls1 mardyn* kernels instead of using *AutoPas*

Besides, logging was not deactivated for *AutoPas* on both machines, which impedes performance even more on HAWK, again, due to the increased number of MPI processes. More detailed analysis and performance tuning on HAWK is subject to the current investigation.

4 Summary and Outlook

We presented recent advances in our particle simulation software *ls1 mardyn*. Automated algorithm selection has been enabled in *ls1 mardyn* via the library *AutoPas*, yielding optimal node-level performance. We further improved parallel performance by incorporating diffusive balancing. The node-level throughput on HAWK suggests very high performance for the three scenarios that we discussed in the future. However, performance gains in the multi-node case are currently not optimal on HAWK, due to the recent porting of our software to this architecture. This requires further performance analysis and more tuning of the number of MPI processes and OpenMP threads per node. Current and future work further focus on techniques to improve auto-tuning in *AutoPas*: testing all combinations of algorithms is tedious and requires many time steps. Therefore, Bayesian and other data analysis methods are being evaluated in this scope.

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. 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, 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. Eckhard et al., 591 TFLOPS multi-trillion particles simulation on SuperMUC, in *Supercomputing. ISC 2013*. ed. by J.M. Kunkel, T. Ludwig, H.W. Meuer. Lecture Notes in Computer Science. (Springer, Berlin, Heidelberg, 2013), pp. 1–12. https://doi.org/10.1007/978-3-642-38750-0_1
3. F. Gratl, S. Seckler, N. Tchipev, H.-J. Bungartz, P. Neumann, AutoPas: auto-tuning for particle simulations. in *2019 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, (2019), pp. 748–757
4. 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)
5. 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)
6. G. Nagayama, P. Cheng, Effects of interface wettability on microscale flow by molecular dynamics simulation. Int. J. Heat Mass Transf. **47**, 501–513 (2004)

7. 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)
8. S. Páll, B. Hess, A flexible algorithm for calculating pair interactions on SIMD architectures. Comput. Phys. Commun. **184**(12), 2641–2650 (2013)
9. D. Rapaport, *The Art of Molecular Dynamics Simulation* (Cambridge University Press, Cambridge, 2004)
10. L. Rekvig, D. Frenkel, Molecular simulations of droplet coalescence in oil/water/surfactant systems. J. Chem. Phys. **127**, 134701 (2007)
11. S. Seckler, F. Gratl, M. Heinen, J. Vrabec, H.-J. Bungartz, P. Neumann, Autopas in ls1 mardyn: massively parallel particle simulations with node-level auto-tuning. J. Comput. Sci. **50**, 101296 (2021)
12. 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)*, (2016), pp. 101–110
13. S. Seckler, F. Gratl, N. Tchipev, M. Heinen, J. Vrabec, H.-J. Bungartz, P. Neumann, Load balancing and auto-tuning for heterogeneous particle systems using ls1 MARDYN. in: *High Performance Computing in Science and Engineering 2019* (2019), To be published
14. N. Tchipev, Algorithmic and Implementational Optimizations of Molecular Dynamics Simulations for Process Engineering (2020), Dissertation
15. N. Tchipev, S. Seckler, M. Heinen, J. Vrabec, F. Gratl, M. Horsch, M. Bernreuther, C.W. Glass, C. Niethammer, N. Hammer, B. Krischok, M. Resch, D. Kranzlmüller, H. Hasse, H.-J. Bungartz, P. Neumann, Twetris: twenty trillion-atom simulation. Int. J. High Perform. Comput. Appl. **33**(5), 838–854 (2019)
16. 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, in *Euro-Par 2015: Parallel Processing Workshops. Euro-Par 2015*. ed. by S. Hunold. Lecture Notes in Computer Science. (Springer International Publishing, Cham, 2015), pp. 774–785. https://doi.org/10.1007/978-3-319-27308-2_62
17. X. Wang, J. Li, J. Wang, X. He, N. Nie, Kernel optimization on short-range potentials computations in molecular dynamics simulations, in *Big Data Technology and Applications. BDTA 2015*. ed. by W. Chen. Communications in Computer and Information Science. (Springer, Singapore, 2016), pp. 269–281. https://doi.org/10.1007/978-981-10-0457-5_25

Project Report on "Load-Balancing for Large-Scale Soot Particle Agglomeration Simulations" (Reprint)

Steffen Hirschmann, Andreas Kronenburg, Colin W. Glass, and Dirk Pflüger

Abstract In this reporting period, we have combined several previous efforts to simulate a large-scale soot particle agglomeration with a dynamic, multi-scale turbulent background flow field. We have built upon previous simulations which include 3.2 million particles and have implemented load-balancing within a versatile simulation software. We have furthermore contributed tests of the load-balancing mechanisms for the agglomeration scenario. We have significantly increased the simulation to 109.85 million particles, superposing short-ranged MD with a dynamically changing multi-scale background flow field. Based on extensive software enhancements for the molecular dynamics software ESPResSo, we have started simulating on the Cray XC40 at HLRS. To verify that our setup reproduces essential physics, we have evaluated load-balancing for a scenario, for which we have scaled down the influence of the flow field to make the scenario mostly homogeneous on the subdomain scale. Finally, we have shown that load-balancing still pays off even for the homogenized version of our dynamic soot particle agglomeration scenario.

Reprinted from Publication Advances in Parallel Computing, Volume 36: Parallel Computing: Technology Trends, Steffen Hirschmann, Andreas Kronenburg, Colin W. Glass, Dirk Pflüger, "Load-Balancing for Large-Scale Soot Particle Agglomeration Simulations", pages 147–156, Copyright 2020, with permission from IOS

S. Hirschmann (✉) · D. Pflüger
Institute for Parallel and Distributed Systems, University of Stuttgart, Universitätsstraße 38, Stuttgart 70569, Germany
e-mail: steffen.hirschmann@ipvs.uni-stuttgart.de

D. Pflüger
e-mail: dirk.pflueger@ipvs.uni-stuttgart.de

A. Kronenburg
Institute for Combustion Technology, University of Stuttgart, Herdweg 51, Stuttgart 70174, Germany
e-mail: kronenburg@itv.uni-stuttgart.de

C. W. Glass
Department of Mechanical Engineering, Helmut Schmidt University Hamburg, Holstenhofweg 85, 22043 Hamburg, Germany
e-mail: glassc@hsu-hh.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_34

Press [1]. The publication is available at IOS Press through http://dx.doi.org/10.3233/APC200035.

1 Introduction

Short-range molecular dynamics (MD) [2] is an important field in Computational Sciences. One particular example of a real-world application is the simulation of soot particle agglomeration, which, for example, is relevant for the efficiency of industrial processes. In these processes, particles collide and link irreversibly. Of particular interest is the morphology of the resulting agglomerates. Because results of a computer simulation allows the examination of morphology of agglomerates over time, computer simulation plays an important role in this area.

The approach we use has been described in [3, 4]: Agglomeration processes are simulated in a precomputed, turbulent flow field. Clustering of particles is driven by Brownian motion as well as the background flow, which gets more important as the particle density decreases. The influence of a turbulent background flow field is of particular interest in particle agglomeration simulations. While small turbulence scales can be resolved in small simulations, the open question remains if large, multi-scale turbulent flows critically influence the results.

In order to get to more realistic agglomeration simulations we use a larger and dynamically changing flow field that covers more scales of turbulence as well as a larger setup including the number of particles. Starting from the largest simulation in [4] ("Case 6"), we increase the domain size by a factor of 3.25. This allows us to cover more realistic scales of turbulence. We keep the original particle loading. Hence, we increase the number of primary particles from 3.2 million in [4] by a factor of $3.25^3 \approx 34.33$ to 109.85 million.

This scenario is very large considering the elaborate physical bonding model used. In fact, to the best of our knowledge it is the largest simulation with ESPResSo so far. And, because we simulate an agglomeration process, the simulation naturally gets more and more heterogeneous over time. Simulations of these sizes and types pose two major challenges: (a) We need large-scale parallelism to cope with a simulation of this size (b) we must dynamically adapt the domain decomposition to the changing particle distribution in order to cope with load-imbalances arising from heterogeneity.

These challenges require us to combine this large-scale real world scenario with our previous efforts to bring dynamic load-balancing to the MD software ESPResSo. In particular, we need to make use of dynamic load-balancing at runtime, the newly created, non-regularly partitioned grids and their associated asynchronous communication [5, 6] as well as other contributions to the MD software at hand, like parallel input and output using MPI-IO.

In order to validate the physical correctness of the scenario and assess the applicability of our load-balancing methods, we use a rather homogeneous version of the scenario. A more homogeneous scenario allows us to first focus on physical correct-

ness of the setup while not crucially depending on the best possible load-balancing. We also test the applicability of our load-balancing methods for this scenario.

The remainder of this work is structured as follows: In Sect. 2 we report on related work. In Section 3 we elaborate on the numerical simulation models as well as the used code and our load-balancing methodology. We describe our simulation setup in Sect. 4. Subsequently, in Sect. 5 we show the physical results and our assessment of load-balancing for this setup. Finally, in Sect. 6 we summarize our work and conclude with a note on further topics to investigate.

2 Related Work

At the core of our modeling are Langevin-based agglomeration processes. These have, e.g., been studied in [7]. We are, however, interested in agglomeration processes that are subject to a turbulent background flow field and that links particles irreversibly. This linking process should completely prohibit rotation and sliding of the particles. To this end, the Langevin-based model has been augmented in [3, 4] to include a coupling to a static flow field as well as dynamic bonding to link particles irreversibly at runtime. In [3] several bonding models that effectively prohibit sliding are proposed and evaluated. We use the so-called "AB" (*all-bonds*) model from this work, which has the advantage, that it does not require the addition of virtual particles.

A different approach to tackle the upscaling of agglomeration simulation is coarse-graining, i.e. aggregating whole clusters into one "super particle" (during the simulation) and, thus, reducing the total computational burden. This is still a field of active research, as the involved modeling is complex. For example, coarse grained particles need to accurately resolve the collision probabilities of the underlying "real" cluster. Algorithmically, this kind of dynamic coarse graining leads to larger cell sizes and, thus, likely to less parallelism and more load-imbalances. Studies and first results for this approach are, e.g., presented in [8]. However, the technique described there is not ready yet for large-scale simulations such as ours.

For load-balancing several heuristics are used in existing MD (and other) software. We have discussed details on the most commonly used ones in [9] and have implemented some of them in the MD software ESPResSo in [5, 6, 10]. In this work, we focus on the partitioner based on space-filling curves (SFC) leveraging the well-known and scalable library `p4est` [11, 12], which in turn uses the Z-curve [13]. The actual partitioning for SFC-based algorithms is performed using so-called chain-on-chain partitioning [14]. Several studies find graph partitioning performs best because of its superior model while SFC-based partitioning is fast and consumes less memory [15, 16]. A more theoretical review of several partitioning algorithms can be found in [17] listing important properties like speed, memory usage, etc. Eibl and Rüde [18] inspect different partitioners for the discrete element method and find that there is a trade-off between scalability and quality of partitioning and recommend an SFC-based strategy for small and mid-sized scenarios. Particularly for MD, several

methods are implemented in the simulation software "ls1 mardyn" and compared in [19–21]. These studies also present cost heuristics to estimate the load of individual subdomains. They give us enough reason to focus on SFC-based partitioning first for our current work.

3 Methodology

Our main methodological approach is two-part. First, we explain what numerical models we use to simulate a soot particle agglomeration process within the molecular dynamics framework. Second, we explain what parallelization and load-balancing approaches we use for the implementation of the numerical models. Additionally, we briefly introduce the relevant quantities for analyzing the shape of agglomerates.

3.1 Numerical Models

We model intermolecular interactions with the well-known Lennard-Jones-12-6 potential which consists of an attractive and a repulsive part,

$$U_{LJ}(r) = 4\epsilon \left(\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right),$$

where σ and ϵ are properties of the modeled primary particles. Particles can be bound together with distance-based and angular harmonic bonds. Their associated potentials are given as

$$U_{distance}(r) = \frac{1}{2} k_h \left(r - r_0\right)^2 \quad \text{, and} \quad U_{angular}(\phi) = \frac{1}{2} k_a \left(\phi - \phi_0\right)^2,$$

where r_0 and ϕ_0 are the equilibrium distance and angle, respectively, and k_h and k_a spring constants. These bonds are established in groups at runtime for each triple of particles that collides. This so-called "AB" model is adapted from [3] and depicted in Fig. 1. Note, that the equilibrium angle ϕ_0 is not constant across bonds but rather different for each one. It is chosen as the angle between the particles at collision time.

In order to model frictional influence from a fluid and Brownian motion, we use Langevin Dynamics. A study of purely Langevin-driven agglomeration processes without turbulent background flow can be found in [7]. The equation of motion is given as:

$$m\ddot{\mathbf{x}} = \mathbf{f} - \gamma \left(\dot{\mathbf{x}} - \mathbf{u}_{flow}(t, \mathbf{x})\right) + \mathbf{R}(t), \tag{1}$$

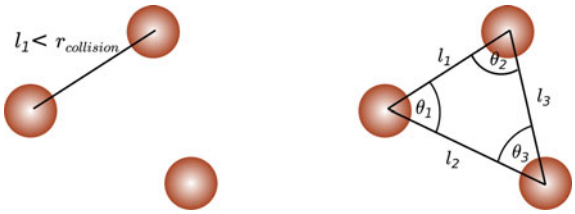


Fig. 1 Visualization of bonding at runtime. Left: If two particles are closer than the collision distance, the algorithm aims to finds a third particle within the collision distance to establish a triangular bonding structure. Right: The bonding structure consists of three angular bonds (indicated as $\theta_1, \theta_2, \theta_3$) as well as three distance-based bonds (indicated as l_1, l_2, l_3) between the particle pairs. If no third particle could be found, only the distance bond l_1 is created

where $\mathbf{f}$ are the forces given by the intermolecular potentials described above. Additionally, $\mathbf{R}(t)$ is a random noise, which, together with the frictional term $\gamma\ (\dot{\mathbf{x}} - \mathbf{u}_{flow}(t, \mathbf{x}))$ models Brownian motion and the temperature, as well as the frictional influence of the fluid. The velocities $\mathbf{u}_{flow}(t, \mathbf{x})$ stem from the fluid (external flow field) at time t and position $\mathbf{x}$. Analogously to [4] we model the friction between the fluid and the particles by Stokes' law, so $\gamma = 3\pi\mu\sigma/C_c$ where μ is the viscosity of the fluid, σ the particle diameter and C_c being the Cunningham correction factor. In ESPResSo, Langevin Dynamics is implemented with a Velocity Verlet integrator, see e.g. [22], combined with a so-called Langevin thermostat [23] that applies the frictional and random forces given the temperature and γ.

3.2 Parallelization and Load-Balancing

We use the simulation software ESPResSo[1] [23, 24], which covers all the relevant physics involved. Relevant parts of the dynamic bonding mechanisms have been implemented in the course of [3, 4] and are also described in [24]. ESPResSo uses the Linked-Cell algorithm [25] in combination with Verlet lists to calculate forces stemming from short-range potentials, like the Lennard-Jones potential, in linear time. Based on the Linked-Cell discretization, ESPResSo defines a uniform spatial domain decomposition to allow for MPI-based parallelization [23]. While the simulation core of ESPResSo is implemented in C++, it exposes a Python-based front-end for setting up the scenario and controlling the simulation [26].

Our adaptions keep the MPI-only parallelization and its 1:1 mapping of subdomains to processes. In [5] we have devised a general scheme to change this fixed decomposition, allowing for arbitrary ones. Based on this work, we have implemented different decompositions. We make the load-balancing mechanism available to the user, so they can conveniently implement strategies for partitioning scenarios in the simulation scripts themselves. Note that this load-balancing mechanism

[1] Extensible Simulation Package for Research on Soft Matter, http://www.espressomd.org.

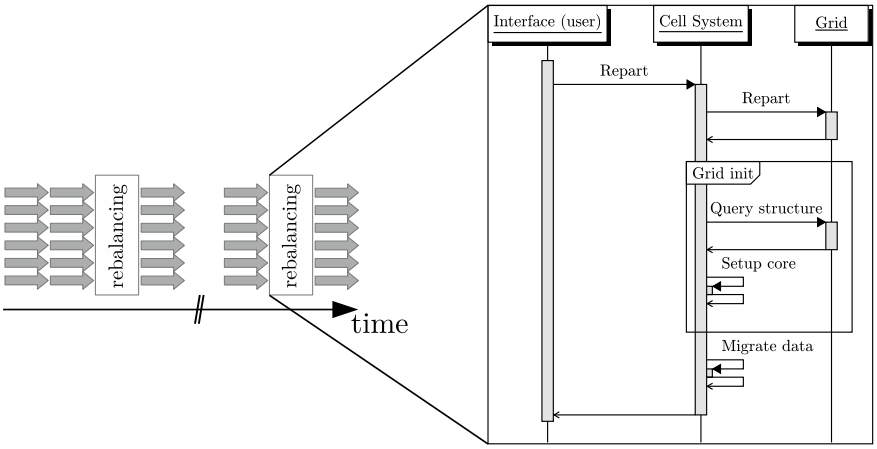


Fig. 2 Left: A sketch of a segment of an MD simulation. The gray arrows depict regular time steps. From time to time the domain decomposition has to be adapted to the underlying scenario (indicated by the "rebalancing" boxes). We outline our implementation on the right: The user calls a function "repart" from their script. The linked cell grid ("cell system") internally asks the grid to repartition itself and then sets up the established partitioning in the core of the simulation software. Afterwards, cell payload (particles) is migrated transparently to the user

is not constrained to the application presented in this work. It can be employed to any heterogeneous simulation in ESPResSo. We depict this ability to do dynamic repartitioning and sketch the underlying implementation in Fig. 2.

Given a function m that defines a suitable load metric or measurement of execution time for every process $p \in \{1, \ldots, P\}$, we partition based on the imbalance $\mathcal{I}(m)$. The imbalance is defined as the maximum over average load measurement: $\mathcal{I}(m) = \frac{P \max\{m(p)\}}{\sum_p m(p)}$. In the current setup, we use $\mathcal{I}(m) > 1.1$ as criterion with the load metric $m(p)$ as the number of particles of process p and partition at most every 1000 time steps. In [5] we have shown for a smaller agglomeration scenario that choosing the number of particles as metric $m(p)$ performs best among a range of different choices.

3.3 Analysis

An important characterization of particle clusters is their fractal dimension D_f [27]. It is the power law relationship of the number of particles to their radius, see e.g. [28], and calculated as

$$N = \left(\frac{r_g}{d}\right)^{D_f},$$

with N the number of particles in the cluster, $d = \frac{\sigma}{2}$ and r_g the radius of gyration, which is the standard deviation of the particle positions $\mathbf{r}_i$ in a cluster:

$$r_g = \sqrt{\frac{1}{N}\sum_{i=1}^{N} ||\mathbf{r} - \mathbf{r}_i||^2} \quad \text{, with} \quad r = \frac{1}{N}\sum_{i=1}^{N} \mathbf{r}_i .$$

4 Simulation Setup

The basic simulation parameters, which will be explained in the following, can be found in Table 1. The reference length, energy and time used for nondimensionalization of the MD simulation can be found in Table 2. The simulation comprises $109.85 \cdot 10^6$ particles, the largest simulation with ESPResSo so far. Each of these primary particles has a diameter of $\sigma = 20\,\text{nm}$. Initially, they are placed in a simulation box of size $l_0 = 2600\sigma$, uniformly randomly distributed. We employ periodic boundary conditions in all dimensions. The particle loading is $\hat{n} = 6.2510^{-3}$. The temperature of the solvent is $T = 600\text{K}$.

The flow field is primarily characterized by its kinematic viscosity ν and its dissipation rate ϵ. Based on these, we can derive the characteristic time scales, namely Brownian diffusion time t_{BM} and the Kolmogorov time scale t_k. These allow us to define the nondimensional Péclet number $\text{Pe} = \frac{t_{\text{BM}}}{t_k}$ that describes the relative importance of turbulence over Brownian motion. The second nondimensional quantity that is used in [4] to describe a scenario is the Knudsen number $\text{Kn} = \frac{l_{mfp}}{\sigma/2}$, where l_{mfp} is the mean free path length of the flow.

Our goal is to reproduce a larger version with more turbulent flow scales of "Case 6" from [4]. This setup uses $\text{Kn} = 11$ and $\text{Pe} = 1$. To achieve that, we generate the external background flow field in a separate pre-processing step using a Direct Numerical Simulation (DNS) of homogeneous, isotropic forced turbulence in a box of length $L = 2\,\pi\,l_0 \approx 326.7\mu\text{m}$. It solves the incompressible Navier-Stokes equations with periodic boundary conditions, discretized on a $64 \times 64 \times 64$ mesh. This mesh is unrelated to the linked cell grid and only defines the resolution of the flow field in the later MD simulation. The viscosity is $\nu = 5.13 \cdot 10^{-5}\text{m/s}^2$ and the dissipation rate $\epsilon = 1.25 \cdot 10^{10}\text{m}^2/\text{s}^3$, which equals the desired values of Kn and Pe. These, however,

Table 1 Basic MD simulation parameters

$\hat{n}$	T	σ	l_0
$6.2510 \cdot \times^{-3}$	600 K	20 nm	$2600\,\sigma$

Table 2 Reference values used for nondimensionalization of physical quantities in the MD simulation

σ	ϵ^*	t^*
20 nm	$1.25\times^{-20}\text{J}$	14 ns

lead to very heterogeneous particle distributions as we have shown in [5, 6] on basis of "Case 6" from [4]. For first experiments at scale, we keep the particle distribution mostly homogeneous on the subdomain scale by reducing the influence of the flow field in the transport equation. Therefore, we scale down $\mathbf{u}_{flow}(t, \mathbf{x}_i)$ in Eq. 1 by about $1/3$. This rescales the gradients of the velocity by an equal factor, and, thus, also the dissipation rate. So in our current simulation the Péclet number is roughly a third of the intended target value.

We compute $6 \cdot 10\times^7$ iterations, each having a length of $dt = 10^{-4}\, t^* = 1.4$fs. Thus, the end of the simulation is at $t_{end} = 8.4\mu$s. The bonding constants $\tilde{k}_a$ and $\tilde{k}_h$ are estimated according to [4] to $\tilde{k}_a = \tilde{k}_h = 1000\epsilon^*$. As collision distance, we use $r_{col} = \sigma$.

As mentioned above, we have extended and use the simulation software ESPResSo as it implements all relevant numerical models, especially dynamic bonding at runtime. We perform the scenario setup as follows: (1) Setup random particles with zero velocities (2) initialize all required potentials (3) equilibrate the system using steepest descend integration [26] (4) reset velocities and forces to zero (5) setup the thermostat as well as collision detection and dynamic bonding (6) start the simulation.

5 Results

One indicator of how much the agglomeration process has progressed in time, is the number of agglomerates of a certain size. We present a histogram of the sizes of the agglomeration for $t_{end} = 600\, t^*$ in Fig. 3. We clearly see that the most prevalent cluster size is about one order of magnitude higher than the smallest possible (2 primary particles). This means that the simulation has left the initial state where the growth of agglomerates is mainly driven by primary particle collisions. At t_{end} the process is primarily driven by cluster-cluster collisions for significant growth of the agglomerates.

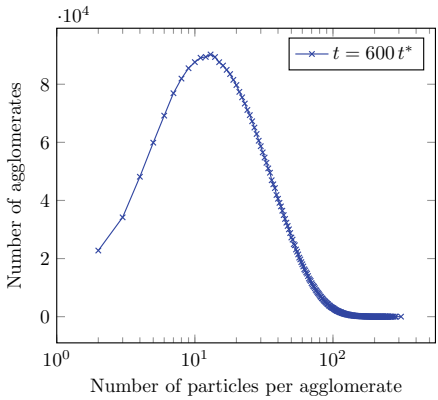


Fig. 3 Histogram of the number of agglomerates per size. Size is in number of particles per agglomerate. The location of the maximum indicates that the agglomeration process has left its initial state where growth is mainly driven by collisions of primary particles

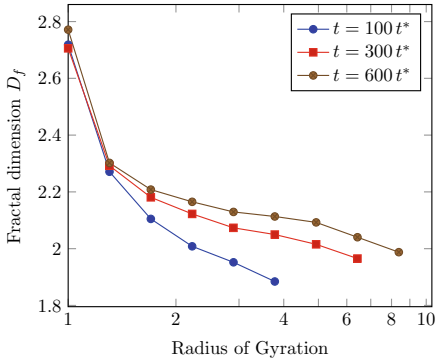


Fig. 4 Average fractal dimension D_f of agglomerates of all agglomerates of a certain radius of gyration at different time steps. While $t = 100\,t^*$ is still early in the simulation, the average D_f does not vary much in later time steps

In Fig. 4 we plot the average D_f of all agglomerates with more than 15 particles at $t = 100\,t^*$, $300\,t^*$, and $600\,t^*$ depending on the radius of gyration of the agglomerates. This relationship enables scientists to understand the agglomeration process and to determine its influence on larger industrial processes and products. Therefore, the criterion for stopping the simulation in [4] is when the individual lines converge. While the process clearly has not converged yet at $t = 100\,t^*$ in Fig. 4, the difference in D_f for the same r_g between $t = 300\,t^*$ and $600\,t^*$ gets significantly smaller, indicating a possible convergence. Since the agglomerates are still rather small ($r_g \leq 10$), the average fractal dimension is in the range of $1.9 \leq D_f \leq 2.2$. These D_f are a bit higher than the ones published by [4] using the same ansatz, as well as experimental data obtained from soot aggregates in (turbulent) flames, see e.g. [29]. However, the smaller the velocities of the superposed, turbulent background flow field in Eq. 1, the higher the influence of Brownian motion. In this case, [4] also states, that with a higher influence of Brownian motion, D_f approaches 2 for larger agglomerates. Beyond that we can see a trend for agglomerates with larger r_g to have a smaller D_f. In conclusion, we can say, that we are able to reproduce physical results for the simulation of soot particle agglomeration, and we assume that the mentioned differences stem from the smaller flow field velocities.

5.1 Load-Balancing

Although we enforce a more or less homogeneous particle distribution among the subdomains, we still use load-balancing to counteract smaller heterogeneity that evolves over time. For unscaled versions of this scenario, good load-balancing is crucial as our studies on the smaller 3.2 million particle setup [5, 6] have shown. Therefore, we also test and show load-balancing results here. The simulation ran on "Hazel Hen" at the High Performance Computing Center Stuttgart (HLRS). It is a Cray XC40 machine with 2 Intel Xeon E5-2680v3 ("Haswell" microarchitecture)

per node, each of which has 12 cores (not counting Simultaneous Multi-Threading) and a Cray Aries interconnect.

We use the snapshot at $t_{end} = 600\,t^*$ for our test. We run the test on 300 nodes, i.e. 7200 processes. The imbalance in the number of particles for a decomposition into equally sized boxes (`MPI_Dims_create`) is about 18.4%, which is quite homogeneous. The runtimes for the default parallelization and our load-balanced one can be found in Fig. 5. We plot the relevant runtimes in the following way: Let

$$
\begin{aligned}
f_1 &= \max\left\{t_{force}(p)\right\}_{p=1}^{P},\\
f_2 &= \max\left\{t_{force}(p) + t_{comm}(p)\right\}_{p=1}^{P}, \text{ and}\\
f_3 &= \max\left\{t_{force}(p) + t_{comm}(p) + t_{int}(p) + t_{sync}(p)\right\}_{p=1}^{P},
\end{aligned}
$$

where "force", "comm", "int" and "sync" refer to the individual components: force calculation, communication, integration and synchronization. Then, we plot f_1, $f_2 - f_1$ and $f_3 - f_2 - f_1$, i.e. the difference of the individual maxima runtimes of the phases. We can see, that despite the homogeneous particle distribution, we achieve a runtime reduction of about 10%. Additionally, in Fig. 6 we can see, that the load-balancing is capable of keeping the imbalance at about the desired level of 1.1 during almost the entire 6 million time steps. The occasional spikes are still being investigated. Given the overall behavior, however, it is likely, that the spikes stem from runtime noise.

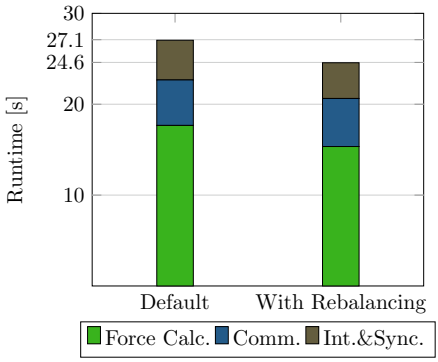


Fig. 5 Runtime of 1000 time steps beginning at $t_0 = t_{end} = 600\,t^*$ for the default, unbalanced paratitioning ("Default") and our SFC-based partitioning ("With Rebalancing") for 7200 processes. For each of the three components (force calculation with the Linked-Cell/Verlet-list method, communication and integration including synchronization) we plot the maximum runtime of any process. Even though the setup is quite homogeneous, we can reduce the runtime by about 10%

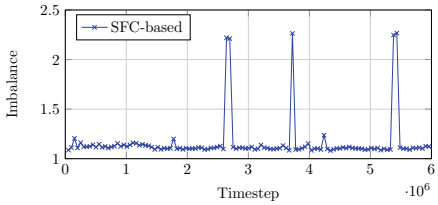


Fig. 6 Imbalance in runtime per 1000 time steps between the different processes. Rebalancing with the SFC-based method is performed if the number of particles diverges by 10% on any process from the average. The number of particles per cell is also used as weights for repartitioning. The spikes are still under investigation. Their occasional occurrences *could* simply be due to an influence of non-deterministic components like hardware, communication, general network load on the HPC system etc

6 Conclusion

We have successfully combined several previous works into one large-scale, dynamically load-balanced soot particle agglomeration simulation. We have set up the simulation with a complex physical bonding model, as well as a dynamically changing, multi-scale turbulent background flow field. We have increased the simulation to over 100 million particles, which is over 30 times larger than results from previous works with ESPResSo and, to the best of our knowledge, the largest simulation ever with ESPResSo.

In order to assess the physical correctness of the new setup, we kept the simulation rather homogeneous by rescaling the velocities stemming from the superposed flow field. This way, we did not have to deal with large heterogeneity, and we were able to reproduce previous results. We have shown that the resulting fractal dimensions of the clusters seem reasonable and consistent with previous results. Also, we have demonstrated, that even when keeping heterogeneity low, load-balancing is still able to reduce the runtimes by about 10% and to consistently keep the imbalance in runtime low throughout the simulation.

6.1 Future Work

There are two paths that we will pursue further. As we have verified that the setup is physically correct and that load-balancing pays off, the first path is to use a physically correct flow field with Pe ≈ 1. This will let us study the impact of the multi-scale turbulent and dynamic background flow field on agglomeration processes at physically relevant scales. We have already simulated a significant fraction of this physically correct simulation. With the end of life of Hazel Hen, we had to pause the simulation, and we intend to continue them as soon as possible on Hawk.

Second, we have implemented different kinds of load-balancing methods in previous work [5, 6, 10], some of which might be more suitable for the heterogeneous version of the simulation. We intend to test different methods as well as different metrics and relate them to scenario properties in order to gain deeper insight into the problem of balancing heterogeneous simulations with a low average fractal dimension of the clusters. We are currently preparing these runs and will start them, once Hawk is stable and fully operational.

Acknowledgements The authors gratefully acknowledge financial support provided by the German Research Foundation (DFG) as part of the former Collaborative Research Center (SFB) 716, and the computing time on "Hazel Hen" granted by the High Performance Computing Center Stuttgart (HLRS). Special thanks go to Rudolf Weeber from the ICP, University of Stuttgart, for his expertise and fruitful discussions.

References

1. S. Hirschmann, A. Kronenburg, C.W. Glass, D. Pflüger. Load-balancing for large-scale soot particle agglomeration simulations. in *Parallel Computing: Technology Trends*, vol. 36, ed. by I. Foster, G.R. Joubert, L. Kucera, W.E. Nagel, F. Peters, *Advances in Parallel Computing*, (IOS Press, March 2020), pp. 147–156. Originally published version of this paper
2. M.P. Allen, D.J. Tildesley, *Computer Simulation of Liquids* (Oxford University Press, Oxford, 1989)
3. G. Inci et al., Modeling nanoparticle agglomeration using local interactions. Aerosol Sci. Technol. **48**(8), 842–852 (2014)
4. G. Inci et al., *Langevin Dynamics Simulation of Transport and Aggregation of Soot Nano-Particles in Turbulent Flows* (Flow, Turbulence and Combustion, 2017), pp. 1–21
5. S. Hirschmann et al., Load balancing with p4est for short-range molecular dynamics with ESPResSo, vol. 32, in *Advances in Parallel Computing*, ed. by S. Bassini et al. (IOS Press, 2017), pp. 455–464
6. S. Hirschmann, C.W. Glass, D. Pflüger, Enabling unstructured domain decompositions for inhomogeneous short-range molecular dynamics in ESPResSo. Eur. Phys. J. Spec. Topics **227**(14), 1779–1788 (2019)
7. L. Isella, Y. Drossinos, Langevin agglomeration of nanoparticles interacting via a central potential. Phys. Rev. E **82**, 011404 (2010)
8. M. Smiljanic et al., Developing coarse-grained models for agglomerate growth. Eur. Phys. J. Spec. Topics **227**(14), 1515–1527 (2019)
9. S. Hirschmann, D. Pflüger, C.W. Glass, Towards understanding optimal load-balancing of heterogeneous short-range molecular dynamics, in *Workshop on High Performance Computing and Big Data in Molecular Engineering* 2016 (HBME 2016), Hyderabad, India, December 2016
10. S. Hirschmann et al., Load-balancing and spatial adaptivity for coarse-grained molecular dynamics applications, in *High Performance Computing in Science and Engineering* ed. by W. Nagel, D. Kröner, M. Resch (Springer, Cham, 2018). https://doi.org/10.1007/978-3-030-13325-2_26
11. C. Burstedde et al., p4est: scalable algorithms for parallel adaptive mesh refinement on forests of octrees. SIAM J. Sci. Comput. **33**(3), 1103–1133 (2011)
12. T. Isaac, C. Burstedde, L.C. Wilcox, O. Ghattas, Recursive algorithms for distributed forests of octrees. SIAM J. Sci. Comput. **37**(5), C497–C531 (2015)
13. G.M. Morton, A Computer Oriented Geodetic Data Base; and a New Technique in File Sequencing. Technical report, IBM Ltd., 1966

14. A. Pınar, C. Aykanat, Fast optimal load balancing algorithms for 1D partitioning. J. Parallel Distrib. Comput. **64**(8), 974–996 (2004)
15. W.F. Mitchell, A refinement-tree based partitioning method for dynamic load balancing with adaptively refined grids. J. Parallel Distrib. Comput. **67**(4), 417–429 (2007)
16. S. Schambeger, J.M. Wierum, Graph partitioning in scientific simulations: multilevel schemes versus space-filling curves. in *International Conference on Parallel Computing Technologies*, vol. 2763, *LNCS*, (Springer, Heidelberg, 2003), pp. 165–179. https://doi.org/10.1007/978-3-540-45145-7_14
17. B. Hendrickson, T.G. Kolda, Graph partitioning models for parallel computing. Parallel Comput. **26**(12), 1519–1534 (2000)
18. S. Eibl, U. Rüde, A systematic comparison of runtime load balancing algorithms for massively parallel rigid particle dynamics. Comput. Phys. Commun. **244**, 76–85 (2019)
19. M. Buchholz, *Framework Zur Parallelisierung von Molekulardynamiksimulationen in Verfahrenstechnischen Anwendungen* (Verlag Dr. Hut, Munich, 2010)
20. M. Buchholz, H.-J. Bungartz, J. Vrabec, Software design for a highly parallel molecular dynamics simulation framework in chemical engineering. J. Comput. Sci. **2**(2), 124–129 (2011)
21. C. Niethammer et al., ls1 mardyn: the massively parallel molecular dynamics code for large systems. J. Chem. Theory Comput. **10**(10), 4455–4464 (2014)
22. B.J. Leimkuhler et al., Integration methods for molecular dynamics. in *Mathematical Approaches to Biomolecular Structure and Dynamics*, ed. by J.P. Mesirov, K. Schulten, D.W. Sumners, The IMA Volumes in Mathematics and its Applications, vol. 82, (Springer, New York, 1996), pp. 161–185. https://doi.org/10.1007/978-1-4612-4066-2_10
23. H. Limbach et al., ESPResSo - an extensible simulation package for research on soft matter systems. Comput. Phys. Commun. **174**(9), 704–727 (2006)
24. A. Arnold et al., ESPResSo 3.1: molecular dynamics software for coarse-grained models, in *Meshfree Methods for Partial Differential Equations* VI, vol. 89, ed. by M. Griebel, M. Schweitzer, Lecture Notes in Computational Science and Engineering, (Springer, Heidelberg, 2013), pp. 1–23
25. R.W. Hockney, J.W. Eastwood, *Computer Simulation Using Particles* (Taylor & Francis Inc., Bristol, 1988)
26. F. Weik et al., Espresso 4.0 - an extensible software package for simulating soft matter systems. Eur. Phys. J. Spec. Topics **227**(14), 1789–1816 (2019)
27. B.B. Mandelbrot, *The Fractal Geometry of Nature*, vol. 173 (W. H, Freeman New York, 1983)
28. R.J. Samson et al., Structural analysis of soot agglomerates. Langmuir **3**(2), 272–281 (1987)
29. U. Köylü, Y. Xing, D.E. Rosner, Fractal morphology analysis of combustion-generated aggregates using angular light scattering and electron microscope images. Langmuir **11**(12), 4848–4854 (1995)

Miscellaneous Topics

Wolfgang Schröder

In the Miscellaneous Topics section the breadth plus the bounty of numerical simulation are evidenced. The simulations in this chapter go beyond fluid mechanics, structural mechanics, thermodynamics, and so forth. This means, another degree of interdisciplinary research is emphasized. The articles clearly show the link between applied mathematics, fundamental physics, computer science, and the ability to develop models such that a closed mathematical description can be achieved which can be solved by highly sophisticated algorithms on up-to-date high performance computers. It is the collaboration of several scientific fields that determines the progress in fundamental and applied research. The subsequent papers, which represent an excerpt of various projects, will confirm that numerical simulations are not only used to compute some quantitative results but to corroborate basic physical new models and to even develop new theories.

In the project of the Institute of Neutron Physics and Reactor Technology of the Karlsruhe Institute of Technology Monte Carlo (MC) codes for nuclear reactor core simulations are used. The remarkable increase of usage in the nuclear community is due to the inherent absence of common approximations over the particle transport modeling. The MC modeling of the particle transport process implies high-computational resource requirements in contrast to the traditional deterministic approach. Therefore, modern MC codes are designed to take advantage of the increasing computational resources and in particular, to be used on massively parallel computer architectures. The EU McSAFE project (Horizon 2020) yields insight, new techniques developments, and coupled codes for multi-physics.

Molecular Modeling and simulation define the project of the Chair of Thermodynamics and Thermal Process Technology of the TU Berlin and the Chair of Thermodynamics of TU Kaiserslautern. Some findings discussed in the joint

W. Schröder
Institute of Aerodynamics, RWTH Aachen University, Wüllnerstr. 5a
52062 Aachen, Germany
e-mail: office@aia.rwth-aachen.de

contribution are a comparison of a micro- and macroscopic solution of a two-phase Riemann problem, a novel equation of state for the bulk viscosity of liquid noble gases, a detailed analysis of the evaporation process of simple fluids, and an analysis of the solid/fluid phase transition for the face centred cubic lattice to name just a few.

Phase-field simulations are considered in the project of the Institute of Applied Materials of the Karlsruhe Institute of Technology and the Institute of Digital Materials of the Karlsruhe University of Applied Sciences. Such simulations support the development of new products. Many of those novel products rely on precisely controlling the properties of their materials. They are closely linked to the inner microstructure of the material which can be predicted by the phase-field method. Using a grand potential approach, results for several processes will be shown. Furthermore, one kind of process will be optimized and the effect on the performance and scalability will be discussed.

Optimum configurations of human skin cell components are investigated in the project of the Chair of High Performance Computing in the Engineering Sciences of the Ruhr University Bochum and the Department of Mathematics of the University Hamburg. Linear elastic mechanical properties of epidermal cell structures including additional constraints for space-filling designs with minimal surfaces are modelled. Several benchmark tests are performed using a gradient-penalized shape optimization algorithm. In addition, an enhanced regularization for locally non-singular deformation mappings is presented for a two-dimensional setting to significantly reduce the number of required optimization steps while retaining very thin inter-cellular channels.

Year 2019 ForHLR II HPC Experience Under the Project "High-Performance Monte Carlo Methods for SAFETY Demonstration-From Proof of Concept to Realistic Safety Analysis and Industry Applications (McSAFE)"

Manuel García, Diego Ferraro, Luigi Mercatali, and Victor Sanchez

Abstract Driven by the growing interest in the nuclear industry in high-fidelity simulations for design and safety analysis of nuclear reactors, the EU Horizon 2020 McSAFE project was set to tackle the implementation of multiphysics tools based on the Monte Carlo particle transport method. The general objective of the project is to improve the prediction of local safety parameters at pin level in Light Water Reactors (LWRs) solving large-scale pin-by-pin depletion and transient problems. This report presents a summary of the simulations performed in the ForHLR II cluster in the framework of the McSAFE project. The HPC resources provided by ForHLR II are critical for the execution of the project, and provide an excellent infrastructure to demonstrate the developed capabilities to the reactor physics community.

1 Background on McSAFE Project Scope and Main Goals

Monte Carlo (MC) codes for nuclear reactor core simulations have experienced a remarkable increase of usage in the nuclear community during past years. One of the main reasons for this interest is the inherent absence of common approximations over the particle transport modeling (such as angular, energy and spatial considerations traditionally done in the standard-industry approach, namely the deterministic codes). Unfortunately, the MC modeling of the particle transport process implies a

M. García (✉) · D. Ferraro · L. Mercatali · V. Sanchez
Institute for Neutron Physics and Reactor Technology (INR) - KIT, Eggenstein-Leopoldshafen, Germany
e-mail: manuel.garcia@kit.edu

D. Ferraro
e-mail: diego.ferraro@kit.edu

L. Mercatali
e-mail: luigi.mercatali@kit.edu

V. Sanchez
e-mail: victor.sanchez@kit.edu

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_35

high-computational resource requirement in contrast with the traditional deterministic approach. To tackle this limitation, the modern MC codes are designed to take advantage of the increasing computational resources, and in particular, to be used under massively parallel computer architectures.

The combination of the MC inherent capabilities to diminish physics approximations together with the use of massive architectures provides the community the capability to perform simulations never expected before.

Under this framework, the EU McSAFE project (Horizon 2020) includes the goal of provide insights, new techniques developments and coupled codes for multi-physics, oriented to tackle the capability of"high-fidelity" core analysis (namely high detailed modeling).

These efforts under McSAFE are intended to provide state-of-the-art tools to cover not only traditional coupled steady state calculations of nuclear reactors cores, but also fast transient analyses and depletion calculations. Therefore, diverse combination of neutronic plus thermal-hydraulic (MC-TH) coupling or neutronic plus thermal-hydraulic (MC-TH) plus fuel-pin mechanics coupling (MC-TH-TM) couplings are to be implemented and tested.

For such purpose diverse coupling strategies and tests were developed between the Serpent 2 Monte Carlo (MC) neutronics, the SUBCHANFLOW (SCF) subchannel TH and the TRANSURANUS (TU) fuel pin mechanic codes, where main insights will be here provided.

Finally, to tackle the resources requirements, all these planned developments should be demonstrated using a high-performance computer (HPC) in order to be able to solve real whole-core rector cores at high detail level (i.e. at pin level resolution). Previous analysis were held during 2018–2019 testing the main codes features within ForHLR II infrastructure [1], where this report will briefly summarize main highlights for the work developed during 2019 period.

2 Year 2019 Experience with ForHLR II Under McSAFE Project

During 2019 several analysis were performed regarding key aspects of massive parallelization of multi-physics Neutronics/Thermal-mechanics/Thermal-hydraulic calculations under McSAFE stated goals. Main insights will be here discussed, where details can be gathered in [2] to [8]. These analysis were done relying in the ForHLR II infrastructure, as far as the inherent computational requirements are not reachable within a standard computational architecture, as already identified in preliminary testing held during 2018 period [1].

For such purpose in-depth testing and ad-hoc developments were held, including (not exhaustive):

- Investigation of multi-physics schemes for neutronic-thermalhydraulic-fuel pin mechanics coupling within diverse standard reactor designs [2, 3].

- Investigation of coupled full scope steady-state MC-based calculations: Verification of coupled steady-state Serpent-SCF within diverse LWR design, such as VVER [4] and PWR [5].
- Investigation of coupled transient MC-based calculations: Preliminary testing [6], Validation of coupled Serpent-SCF transient calculations within PWR-kind geometrical and operational condition, namely using experimental data from SPERT-IIIE reactor [7] and verification of coupled Serpent-SCF transient calculations within a full-core scope [5].
- Development, implementation and testing of Domain-decomposition techniques, oriented to tackle problems previously identified regarding full-core burnup calculations.

A brief discussion of some of these aspects are provided in the following subsections, were in-depth analysis can be found in the listed references and the McSAFE Project reports. During these analyses, the ForHLR II computer cluster was very valuable to move forward in the development of high-fidelity simulations based on Monte Carlo simulations. The obtained results are very promising and hence, a continuation of the HPC-work at the ForHLR II Cluster is essential for the proposed research in the multi-physics and multi-scale field applied to reactor systems.

2.1 Investigation of Multi-physics Schemes: First Results for Neutronic-thermalhydraulics-Fuel Pin Mechanics Coupling

Regarding multi-physics development, a three code coupling using the three codes mentioned above (i.e. Serpent 2, SCF and TU) was developed and implemented to be run within the ForHLR II infrastructure [2, 3]. This high-fidelity scheme was developed using an object-oriented mesh-based implementation approach, as shown in Fig. 1. The calculation scheme is based on replacing the simple fuel-rod model in SCF with the more sophisticated thermomechanic model in TU, with the aim at adding fuel-performance analysis capabilities to the traditional neutronic-thermalhydraulic methodology.

Using this approach two different analysis were developed, namely using VVER [2] and PWR [3] geometries, using the ForHLR II infrastructure to tackle the inherent computational requirements of the proposed approach.

Regarding the first one, to test this three-code system, a 360-day depletion calculation of a VVER-1000 fuel assembly was performed using a pin-by-pin model and the results were compared with the ones obtained with Serpent2-SCF without TU.

Several analysis were held regarding the capabilities of this multi-physics coupling, as shown in Fig. 2 as example for the power distribution evolution with burnup, comparing with results without the fuel pin mechanics physics. Further details can be directly gathered in [2].

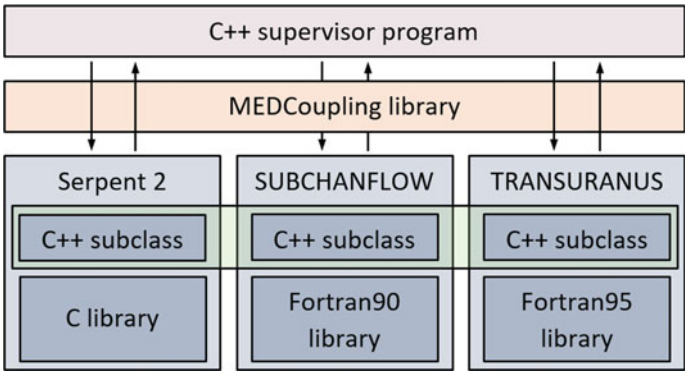


Fig. 1 Three-code coupling scheme (neutronics - TH - fuel pin mechanics) [2]

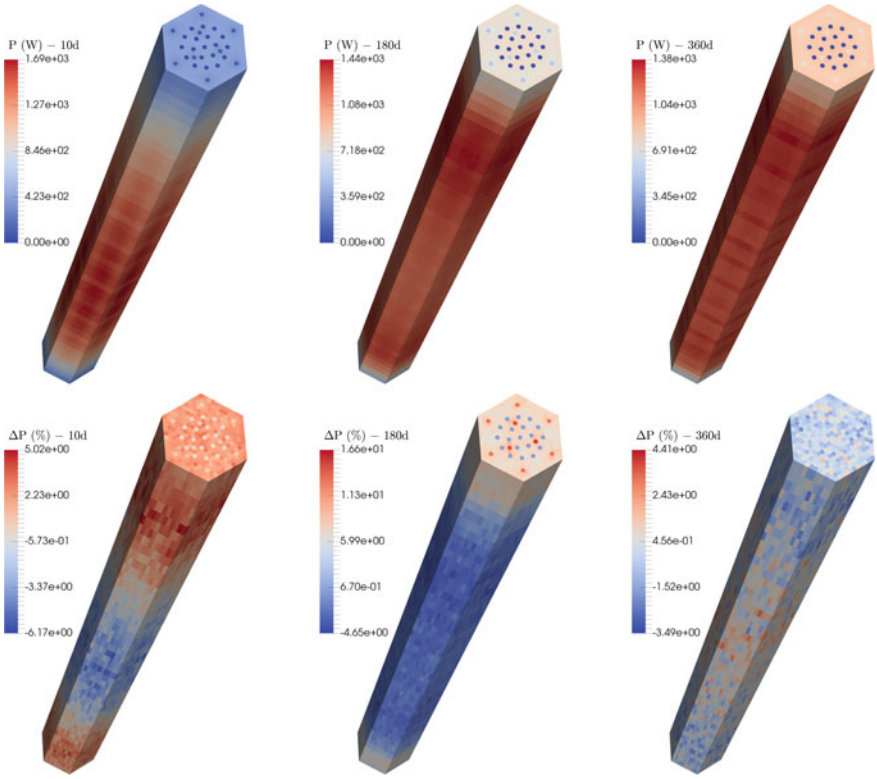


Fig. 2 Serpent2-SCF-TU power (top) and difference with Serpent2-SCF (bottom) [2]

Regarding PWR geometries, the same procedure was developed within PWR fuel assembly using as basis a benchmark that consists of a Westinghouse 17 × 17 fuel assembly with 24 guide tubes and a single instrumentation tube, defined at Beginning of Cycle (BOC) and Hot Full Power (HFP) in the benchmark, as shown in Fig. 3.

Again, diverse pin-level results were analyzed and compared with those without fuel pin-mechanics (i.e. 2-code coupling), as shown is Fig. 4 as an example. Further details can be directly gathered in [3].

The analysis of the results of both example cases were shown to be consistent, which serves as verification of the implementation of the three-code coupling. To summarize, the general implementation of Serpent2-SCF-TU has been verified, though there still remain issues to analyze, namely the treatment of the Doppler feedback and the influence of high-burnup effects that could lead to larger differences between the fuel models of SCF and TU.

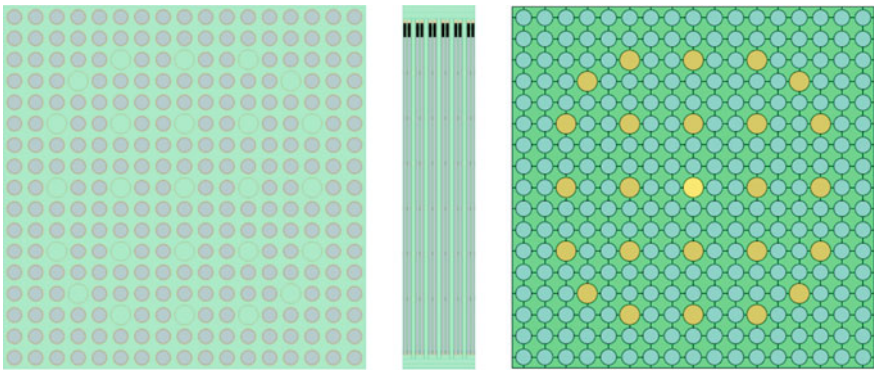
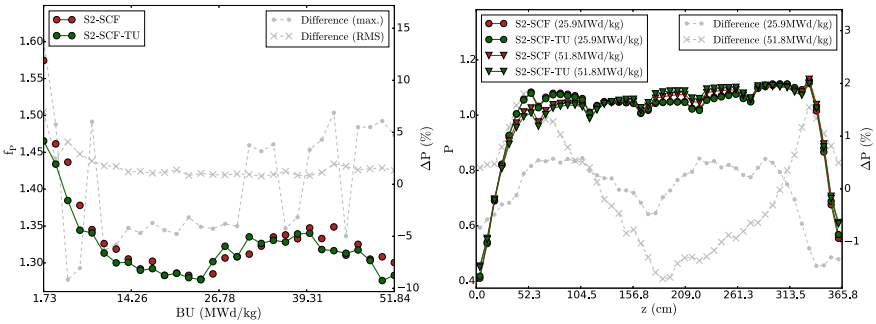


Fig. 3 Serpent2 and SCF (right) models for PWR geometries [3]



(a) Power peak factor.

(b) Axial power profile.

Fig. 4 Examples of results analyzed within PWR geometries [3]

2.2 *Investigation of Coupled Full Scope Steady-State MC-Based Calculations: Verification of Coupled Steady-State Serpent-SCF Within Diverse LWR Design, Such as VVER and PWR*

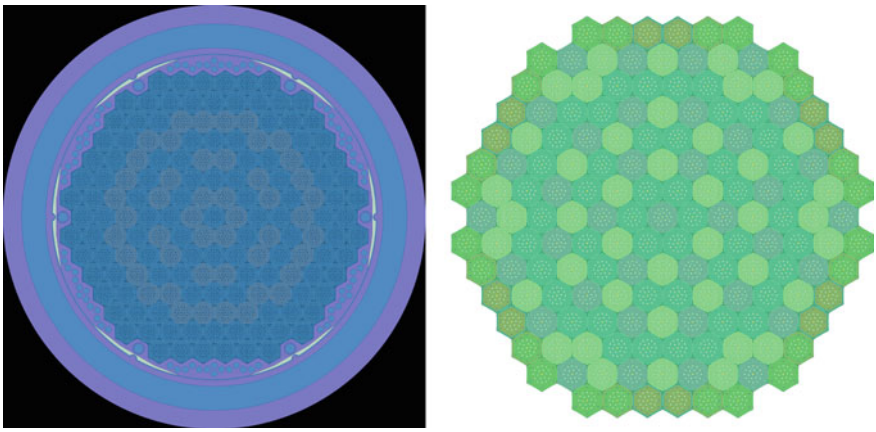
Regarding the full-scope approach, the coupling capability was tested for a series of full-core cases. For such purpose, a coupled master-slave approach between the MC Serpent 2 and SCF codes was considered using a hybrid MPI-OMP approach. Tests within diverse geometries and operational conditions were held, where details can be gathered [4, 5].

Firstly, a full scope pin-by-pin modeling was developed for a VVER benchmark that provides experimental results for diverse reactor states [4]. As a result, two different reported states from first operating cycle were calculated with this high-fidelity tool. Main models are shown in Fig. 5 both for Serpent and SCF, where further details can be found in [4].

Main aspects were verified with these models, where the ForHLR II resources were mandatory to tackle the inherent requirements. Diverse full-scope results were obtained for this VVER case, as shown in Fig. 6.

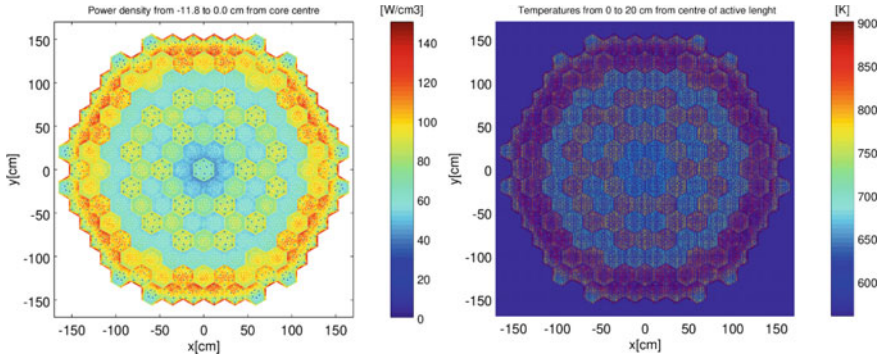
Afterwards, full-scope coupled steady-state calculations were developed for a well-known PWR benchmark [5], where again the use of an HPC was mandatory to tackle the computational requirements. As for the VVER case, the main Serpent and SCF geometries are shown in Fig. 7, where example highly detailed results are shown in Fig. 8.

Regarding computational resources, about 7e4 CPU mins (@ 2.60GHz Intel(R) Xeon(R) CPU E5-2660 v3) were required for each N-TH iteration step for these full



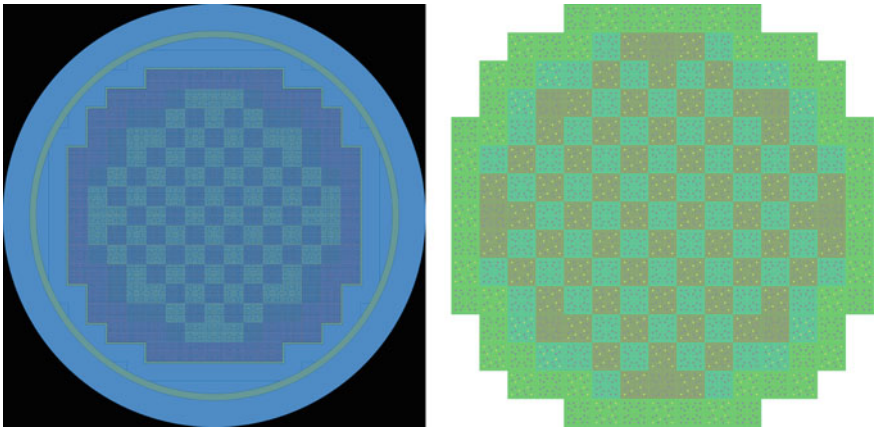
(a) Serpent model. (b) SCF model.

Fig. 5 Examples of Serpent2 and SCF models for a full-core VVER [4]



(a) Power density. (b) Temperature.

Fig. 6 Examples of Serpent2 and SCF results for a full-core VVER [4]

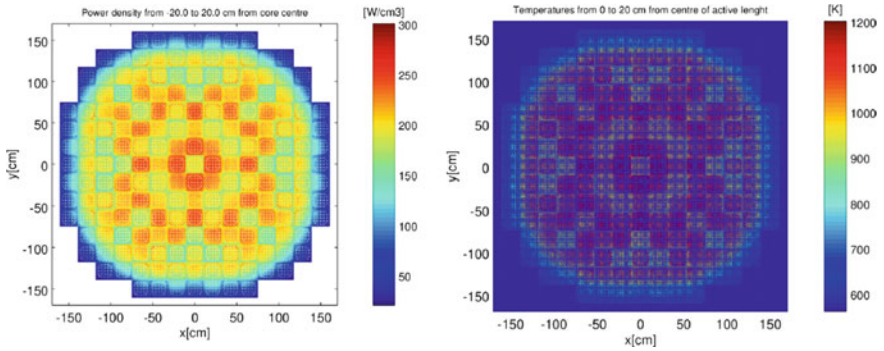


(a) Serpent model. (b) SCF model.

Fig. 7 Examples of Serpent2 and SCF models for a full-core PWR [5]

core pin-by-pin coupled calculations, thus reinforcing the importance of the ForHLR II architecture.

The capabilities to develop full-core calculations with coupled N-TH for both PWR and VVER real geometries and operational conditions were assessed [4, 5]. Besides the global agreement, pin-level detailed (i.e. high-fidelity) results for the interchanged fields were also analyzed, showing both the capabilities available within this approach and the consistency of the behavior of these parameters.



(a) Power density.

(b) Temperature.

Fig. 8 Examples of Serpent2 and SCF results for a full-core PWR [5]

2.3 Investigation of Coupled Transient MC-Based Calculations

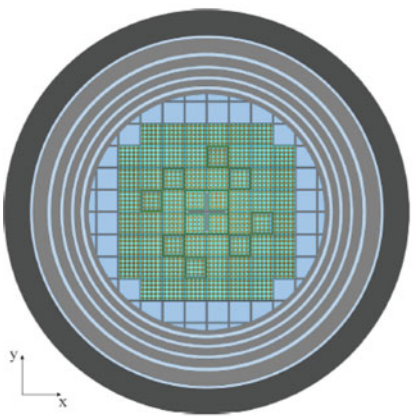
One of the most novel applications of the tools developed within the McSAFE project is the pin-by-pin coupled calculations of transient scenarios. These analyses represent the most demanding resources, as already identified in preparatory works [1]. In this sense, during 2019 period both a preliminary testing [6], a validation of the coupled Serpent-SCF transient calculations within PWR-kind geometrical and operational condition [7] and full-scope PWR verification [7] were held.

Regarding the validation efforts, several scenarios from the experimental campaign held within the SPERT-IIIE reactor were modeled and high-fidelity results were obtained [7]. These experiments represent RIA-kind (reactivity insertion accidents) scenarios, where a sudden withdrawal of a control rod is considered and a transient calculation is held. Both Serpent and SCF models are presented in Fig. 9, where examples of results are presented in Fig. 10 (details can be found in [7]).

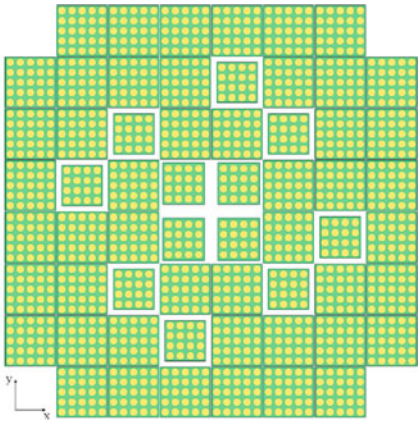
The level of agreement with experimental results was found to be very good, where again the computational resources were prohibitive without an HPC infrastructure, as far as 10e3 CPU hours where required for the complete 1s transient (using an hybrid MPI/OMP implementation within 500 MPI nodes in ForHLR II) [7].

In view of the scaling of these novel applications to full-core PWR calculations (which represents a potential industry-like application), a RIA-kind coupled transient scenario was developed, using the same PWR benchmark from the steady-state calculations [5] in last section, where a CR is suddenly withdrawn. Main obtained results were obtained and compared with reference ones, as show in Fig. 11, where details can be found in [5].

It is important to note that main aspects for this full-scope transient case were correctly reproduced, showing the aptness of the approach for full-scope transient calculations. On the contrary, the computational requirement arises as a limiting

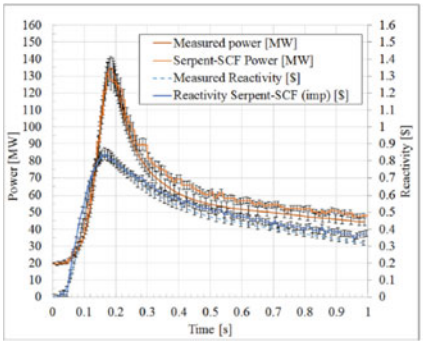


(a) Serpent model.

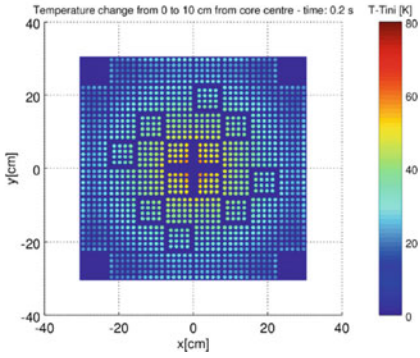


(b) SCF model.

Fig. 9 Examples of Serpent2 and SCF models for the SPERT IIIE experiments [7]



(a) Global power and reactivity comparison with experimental results.



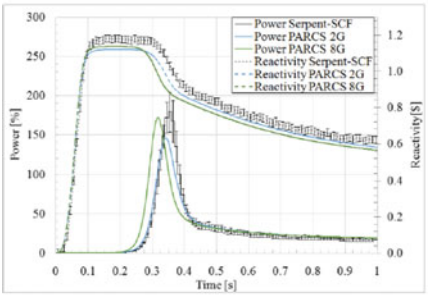
(b) Change in temperature distribution.

Fig. 10 Examples of Serpent2 and SCF results for the SPERT IIIE experiments [7]

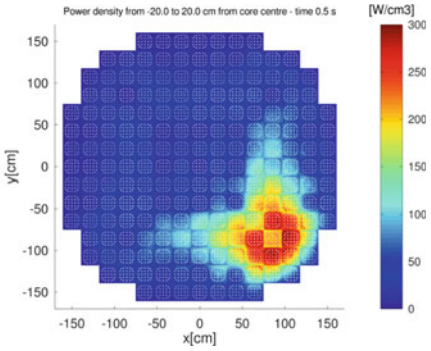
factor for such problems, where up to 1e5 CPU mins, that represented about 122 h using 1280 processors in the ForHLR II, were necessary for the whole 1 s calculation.

2.4 Memory Scalability in Burnup Calculations

Previous sections showed the aptness of this novel approach to develop high-fidelity analysis within real applications in reactor physics both for steady-state and transient

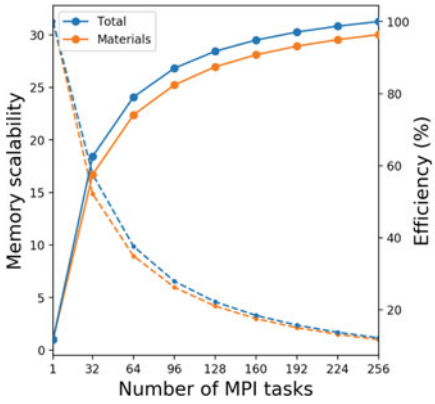


(a) Global power and reactivity comparison with experimental results.

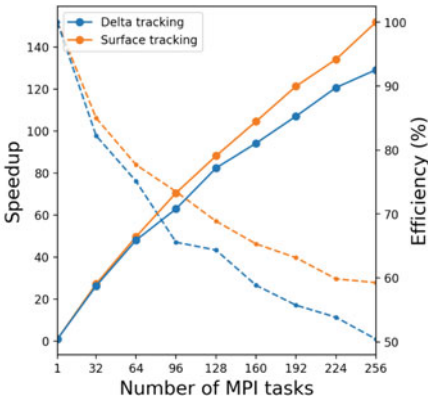


(b) Power distribution.

Fig. 11 Examples of Serpent2 and SCF results for a PWR benchmark [5]



(a) Memory scalability.



(b) Speedup.

Fig. 12 Main results for the CDD implementation

calculations. In spite of this, when dealing with burnup calculations, limitations related to RAM memory requirements were identified in previous periods [1].

In this framework, a Collision-based Domain Decomposition (CDD) scheme has been implemented in Serpent2 to tackle problems with memory footprints that exceed the typical in-node memory of HPC systems [8], in particular the ForHLR II cluster. Dividing the material data of the target problem across MPI tasks, memory scalability is achieved and nodes can be added to fit the problem in memory. This is not possible in the traditional particle-based parallel scheme used in Monte Carlo neutron transport, which relies on domain replication and has no memory scalability.

This new method has been tested in the ForHLR II cluster with the memory scalability and speedup shown in Fig. 12. These results correspond to hybrid MPI-OpenMP execution with 20 OpenMP threads per MPI task. The memory scalability

is quite good and allows for the simulation of huge systems in terms of memory demand. The speedup is very good as well, with almost 50% efficiency at 256 MPI tasks (5120 cores).

3 Conclusions and Foreseen Steps

The unique possibility to use the KIT/SCC ForHLR II computer cluster was very valuable to move forward in the development of high-fidelity simulations based on Monte Carlo simulations. The obtained results showed the aptness of the proposed approach within the McSAFE for the intended goals. Further works related to full scope calculations including burnup are foreseen as final step, still under development, as briefly discussed within last section.

As a result, the series of analyses developed during 2019 period relying on the ForHLR II architecture provided insights and new techniques developments oriented to tackle the novel high-fidelity reactor core analysis, thus representing several interesting industry-like applications.

References

1. D. Ferraro et al., FH2 HPC Experience under the project 'high-performance monte carlo methods for safety demonstration-from proof of concept to realistic safety analysis and industry applications (McSAFE) (2018)
2. M. García et al., A Serpent2-SUBCHANFLOW-TRANSURANUS coupling for pin-by-pin depletion calculations in Light Water Reactors. Ann. Nucl. Energy **139**, 107213 (2020)
3. M. García et al., SERPENT2-SUBCHANFLOW-TRANSURANUS PIN-BY-PIN DEPLETION CALCULATIONS FOR A PWR FUEL ASSEMBLY, in *Proceedings of the PHYSOR2020 Conference* (2020)
4. D. Ferraro et al., Serpent/SUBCHANFLOW coupled calculations for a VVER core at hot full power, in *Proceedings of the PHYSOR2020 Conference* (2020)
5. D. Ferraro et al., OECD/NRC PWR MOX/UO2 core transient benchmark pin-by-pin solutions using Serpent/SUBCHANFLOW. Ann. Nucl. Energy **147**, 107745 (2020)
6. D. Ferraro et al., Serpent/SUBCHANFLOW pin-by-pin coupled transient calculations for a PWR minicore. Ann. Nucl. Energy **137**, 107090 (2020)
7. D. Ferraro et al., Serpent/SUBCHANFLOW pin-by-pin coupled transient calculations for the SPERT-IIIE hot full power tests. Ann. Nucl. Energy **142**, 107387 (2020)
8. M. García et al., A collision-based domain decomposition scheme for large-scale depletion with the serpent 2 monte carlo code. Ann. Nucl. Energy **152**, 108026 (2021)

Molecular Modeling and Simulation: Model Development, Thermodynamic Properties, Scaling Behavior and Data Management

Matthias Heinen, René S. Chatwell, Simon Homes, Gabriela Guevara-Carrion, Robin Fingerhut, Maximilian Kohns, Simon Stephan, Martin T. Horsch, and Jadran Vrabec

Abstract We are outlining our most recent findings, covering: 1) A comparison of a micro- and macroscopic solution of a two-phase Riemann problem obtained from molecular dynamics simulations and finite volume schemes; 2) A novel equation of state for the bulk viscosity of liquid noble gases based on a multi-mode relaxation ansatz; 3) A detailed analysis of the evaporation process of simple fluids; 4) Diffusion coefficients of quaternary liquid mixtures obtained with the Green-Kubo formalism; 5) An analysis of the solid/fluid phase transition for the face centered cubic (fcc) lattice; 6) The relative permittivity of mixtures of water and acetone; 7) An assessment of the reliability and reproducibility of molecular simulation results; 8) Techniques for the data management in simulation workflows, including annotations of simulation outcomes with appropriate metadata standardized by an ontology.

1 Two-Phase Shock Tube Scenario

Large scale molecular dynamics (MD) simulations of a two-phase shock tube scenario were conducted. These simulations were intended to serve as a benchmark for macroscopic solutions obtained from computational fluid dynamics (CFD) simulations employing finite volume (FV) schemes. Two macroscopic approaches were

M. Heinen · R. S. Chatwell · S. Homes · G. Guevara-Carrion · R. Fingerhut · J. Vrabec (✉)
Fachgebiet Thermodynamik und Thermische Verfahrenstechnik, Technische Universität Berlin, Ernst-Reuter Platz 1, 10587 Berlin, Germany
e-mail: vrabec@tu-berlin.de

M. Kohns · S. Stephan
Lehrstuhl für Thermodynamik, Technische Universität Kaiserslautern, Erwin-Schrödinger-Straße 44, 67663 Kaiserslautern, Germany

M. T. Horsch
School of Psychology and Computer Science, University of Central Lancashire, 45 Fylde Rd, Preston PR1 1JN, Preston, UK

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_36

considered: the homogeneous equilibrium method (HEM) and the sharp interface method. Both are implemented in the discontinuous Galerkin spectral element method (DGSEM) framework *FLEXI* [24].

In contrast to the scenario in Ref. [25] that considered two supercritical states, known as the classical Riemann problem, the present scenario consisted of a liquid and a vapor phase, connected through a planar interface. The thermodynamic states of the liquid and vapor phases were specified such that they were out of equilibrium and hence phase transition occurred. This scenario is known as two-phase Riemann problem. While the classical Riemann problem is fully understood, the two-phase Riemann problem has implications for the system of equations that cannot be solved in a straightforward manner.

As in Ref. [25], the Lennard-Jones Truncated and Shifted (LJTS) fluid was assumed for two reasons: it is computationally cheap in MD simulations and an accurate equation of state (EOS) [21] is available for the macroscopic solution.

The initial configurations of the MD simulations were prepared in two steps, cf. Fig. 1. First, a vapor-liquid equilibrium (VLE) was maintained at a temperature of

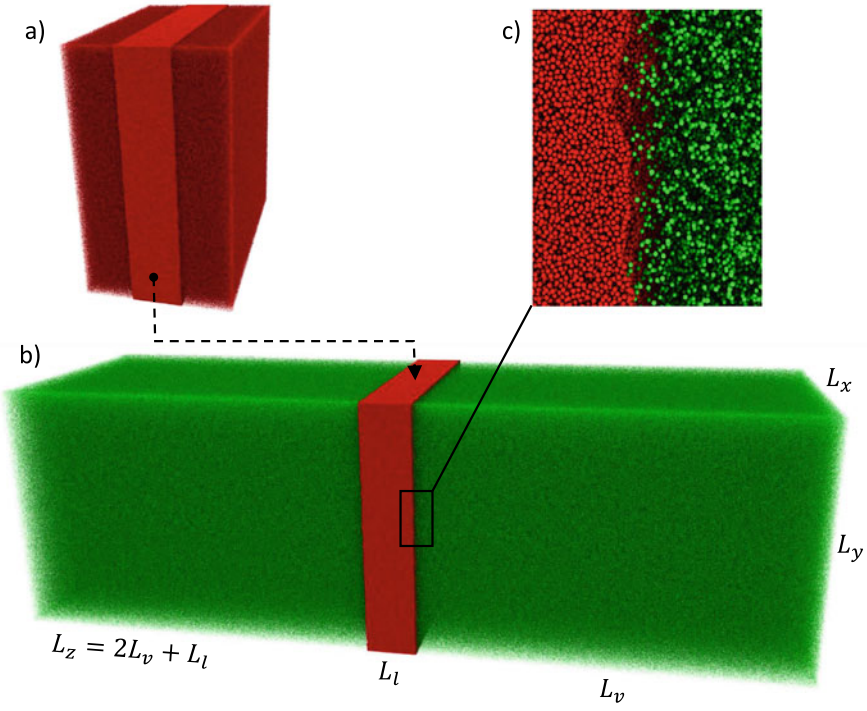


Fig. 1 Snapshots of the prepared molecular systems, rendered with the cross-platform visualization prototyping framework MegaMol [18] and the Intel OSPRAY plugin. a) Final configuration of the vapor-liquid equilibrium simulation from which the liquid phase was extracted to build the test case scenarios. b) One of the test case scenarios with a vapor phase (green) diluted to 70% of the saturated vapor density at the temperature $T = 0.8$. c) Close-up look at the interface

$T = 0.9$ in the usual reduced units. The liquid phase was extracted and brought into contact with a vapor phase at a lower temperature $T = 0.8$ and a lower density in a symmetric setup. Three cases were considered with varying the density of the vapor phase ρ_v, i.e. specifying 50%, 70% and 90% of the saturated vapor density at a temperature of $T = 0.8$. The system dimensions, specified in particle diameters σ, were defined as follows: The extent of the vapor phase $L_v = 1500$ was chosen to be wide enough so that the shock wave exerted from the liquid phase, which is a consequence of the global non-equilibrium, could be observed for a sufficiently long time period before it reached the periodic boundary. The specified width of the liquid phase $L_l = 200$ was a compromise between being sufficiently wide such that the opposite vapor-liquid interfaces do not interfere with each other, through rapid state changes due to evaporation, and being small enough to keep the computational cost on an acceptable level. To check whether the specified width of L_l is appropriate, an exemplary simulation was repeated with a doubled width $L_l = 400$. From that, almost identical results were obtained.

The cross-sectional area had an extent of $10^6\sigma^2$ so that the sampling of the rapidly changing profiles, i.e. temperature, density and hydrodynamic velocity, employing a classical binning scheme with a high spatial and temporal resolution yielded an excellent statistical quality. Because of the large number of up to $N \approx 3 \cdot 10^8$ particles, all simulations were carried out with the massively parallel code *ls1 mardyn* [42]. This code is being continuously improved and was recently optimized with respect to its node-level performance and parallel efficiency [56].

Results for the case with 50% of the saturated vapor density are shown in Fig. 2. While the HEM approach is not able to reproduce the results of MD simulation, the sharp interface approach showed a very good agreement with the MD data in the homogeneous bulk phases, except for the vicinity of the interface. It reproduced the propagation speed of the shock wave and the characteristic shapes of all profiles. Even the magnitudes of the profile plateaus were quantitatively matched.

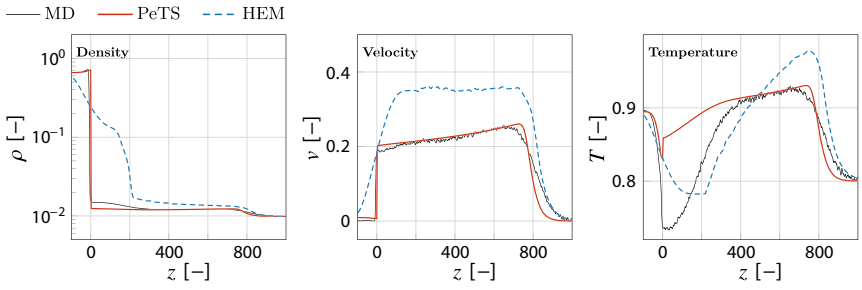


Fig. 2 Results for the case with a vapor density of 50% of its saturation value. The plots show profiles of density, velocity and temperature at the time instance $t = 200$ obtained from the sharp interface method, HEM and MD

2 Bulk Viscosity of Liquid Noble Gases

Stokes' hypothesis postulates that any form-invariant changes of a fluid's local volume, i.e. compression or dilatation, are not associated with the dissipation of linear momentum, which is synonymous with a vanishing bulk viscosity $\mu_b = 0$. Despite theoretical and experimental evidence to the contrary, this hypothesis is still widely applied throughout all branches of fluid mechanics. The bulk viscosity of liquid noble gases was studied here on the basis of a multi-mode relaxation ansatz and an equation of state is proposed [9]. The relaxation ansatz is based on a large data set that was generated by dedicated atomistic simulations, resulting in an EOS exposing the bulk viscosity as a two-parametric power function of density, with the parameters being functions of temperature. The noble gases' atomistic description rests on the Lennard-Jones potential.

The Green-Kubo formalism relates the bulk viscosity to time-autocorrelation functions (ACF) that were sampled in the microcanonical (*NVE*) ensemble, utilizing the fully open source program *ms*2 [44]. To reduce finite size effects and to gain better statistics, ensembles containing $N = 4096$ particles were placed in cubic volumes with periodic boundary conditions. To resolve the small-scale pressure fluctuations, a small integrator time step was used and each autocorrelation function was sampled over a substantial time period. Relatively large ensemble sizes in combination with long simulation times are computationally demanding and thus require modern HPC architectures. The bulk viscosity can be determined microscopically by ACF of local small-scale, transient pressure fluctuations that are intrinsic in any fluid under equilibrium. These pressure fluctuations have been observed to relax in different modes. Each mode decays exponentially over time, following a Kohlrausch-Williams-Watt function. For liquid noble gases, three superimposing relaxation modes were found to be present, leading to the relaxation model

$$B_R(t) = C_f \exp\left(-\left(\frac{t}{\delta_f}\right)^{\beta_f}\right) + C_m \exp\left(-\left(\frac{t}{\delta_m}\right)^{\beta_m}\right) + C_s \exp\left(-\left(\frac{t}{\delta_s}\right)^{\beta_s}\right) . \quad (1)$$

The first term describes the fast, and the subsequent terms the intermediate and slow modes, respectively. The weighting factors are constraint, i.e. $C_f + C_m + C_s = 1$, and the Kohlrausch parameters δ_i, β_i are a measure of relaxation time scale and distortion from the exponential function. The model's eight independent parameters $C_f, C_m, \delta_f, \delta_m, \delta_s, \beta_f, \beta_m, \beta_s$ were determined by fitting the relaxation model B_R to the data sampled by MD. Each mode's average relaxation time τ_i is defined as integral mean value of its respective relaxation model contribution $B_{R,i}$. As originally proposed by Maxwell, the bulk viscosity μ_b is proportional to the cumulative averaged relaxation time

$$\mu_b = K_r \sum_{i=1}^{3} \lim_{t\to\infty} \int_0^t dt \left(B_{R,i}\right) , \quad (2)$$

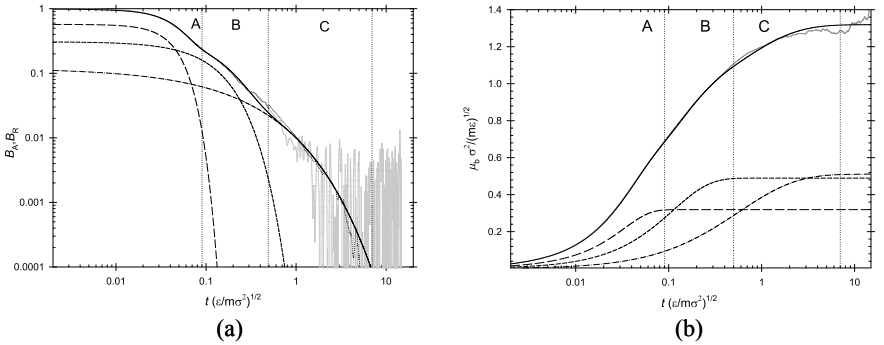


Fig. 3 (a) Comparison of a sampled ACF with the relaxation model including all three modes. While the gray line constitutes the sampled ACF, the solid black line represents the relaxation model and its fast (dashed), intermediate (short-dashed) and slow (dashed-dotted) modes, respectively. (b) Comparison of the integrated sampled autocorrelation function with the relaxation model. Due to noise contributions to the slow mode of the sampled autocorrelation function, the bulk viscosity is difficult to determine precisely by molecular dynamics simulation (gray line). In contrast, the employed relaxation model (solid black line) converges towards an unambiguous value at finite times. The dashed, short-dashed and dotted-dashed lines represent the fast, intermediate and slow modes, respectively

with the proportionality constant K_r being the fluid's relaxation modulus. The sampled ACF partitions into three segments, with each segment being dominated by a different mode, cf. Fig. 3a. In contrast to the sampled ACF that is plagued by noise, the employed relaxation model's time integral properly converges to a definite value, thus allowing to determine μ_b unambiguously, cf. Fig. 3b.

Applying the relaxation ansatz to all sampled state points generates a large dataset from which the EOS emerges as a two-parametric power function with both parameters showing a conspicuous saturation behavior over temperature. After passing a temperature threshold, the bulk viscosity is found to vary significantly over density, a behavior that resembles the frequency response of a one pole low-pass filter.

3 Evaporation of Simple Fluids

Evaporation phenomena play a crucial role in process engineering and in many other fields, but they are not yet fully understood. In order to gain further knowledge about these basic phenomena, MD simulations were conducted building upon Ref. [23]. The simple and computationally efficient LJTS potential was applied to describe the interactions between the particles. A parallelepiped was employed as simulation volume, consisting of one liquid and one vapor phase, respectively. A net evaporation flux was constrained by establishing a vaccuum boundary condition in the vapor. Particles reaching this vaccuum region were deleted. In order to maintain a constant number of particles N in the simulation domain, new particles had to be added.

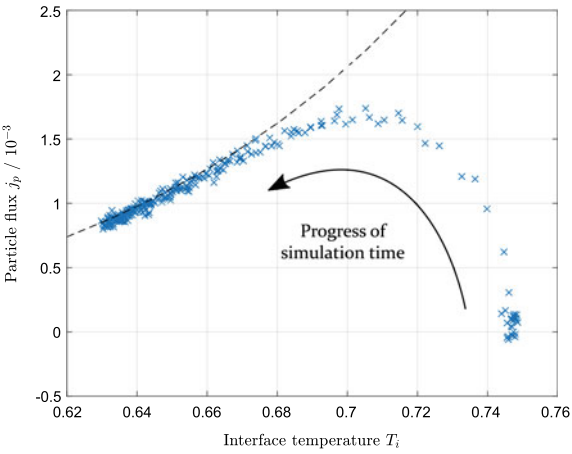


Fig. 4 Particle flux j_p over the interface temperature T_i over the course of one simulation. Throughout the simulation run, T_i declined until stationarity was reached. The particle flux first increased up to a maximum, then decreased, until it finally converged to a stationary value. The dashed line marks a fit through the respective stationary values of all simulations

This was achieved by a method proposed in Ref. [22]. Numerous simulations were conducted for a range of temperatures of the bulk liquid as well as varying distances between the bulk liquid and the interface.

The systems under investigation contained between 1.3 and $8.3 \cdot 10^6$ particles. The utilized MD software was *ls1 mardyn* [42], which is designed for massive parallelization so that up to $7.8 \cdot 10^3$ cores could be used for one simulation run.

It was found that the evaporation flux setting in led to a decrease of the interface temperature T_i. Since the particle flux j_p depends on T_i, a decreasing temperature induces a drop of j_p itself. Upon progress of simulation, the interface temperature and the particle flux converge towards a limit, cf. Fig. 4. Their value depends on the bulk liquid temperature among others. For all conducted simulations, the respective stationary values can be expressed by the dashed line in Fig. 4.

4 Diffusion in Quaternary Liquid Mixtures

Most mass transfer processes occurring in nature and in technical applications involve liquid solutions with more than three components. Rate-based methods employed for modeling, design and control of separation unit operations in chemical engineering, such as distillation, rely on mass and energy transfer models which require reliable information on diffusion coefficient data for the regarded mixtures. Therefore, there is a significant interest in the improvement of experimental methodologies and the development of reliable methods for the prediction of mutual diffusion coefficients of liquid multicomponent mixtures.

Fick's law for diffusion in a quaternary mixture requires nine different diffusion coefficients that depend on temperature, pressure, composition and the regarded frame of reference. The presence of six cross diffusion coefficients makes

interpretation and data processing in experimental work a challenging task which often leads to large experimental uncertainties. Despite the continuous improvement and development of experimental techniques during the last decades, the availability of diffusion coefficients of mixtures containing four components is still very poor. Thus, the growing need of accurate diffusion data for basic research and engineering applications cannot be satisfied by experimental measurements alone.

Most predictive equations for multicomponent diffusion of liquids rely on extensions of the Darken relation [3] and are therefore only truly valid for ideal mixtures. The underlying physical phenomena in non-ideal mixtures are still not well understood and the lack of data impedes the development and verification of new predictive equations. In this context, MD offers an alternative path not only to assess multicomponent diffusion coefficients, but also to gain insight into the underlying microscopic behavior.

Recently, the ability of MD to predict the Fick diffusion coefficient matrix of a quaternary liquid mixture has been demonstrated for water + methanol + ethanol + 2-propanol [19]. However, because of the lack of experimental data, only consistency test could be performed for the predicted diffusion data. Here, the Fick diffusion coefficient matrix of the mixture cyclohexane + toluene + acetone + methanol, which was recently studied with Raman spectroscopy [43], was successfully predicted solely with molecular simulation techniques.

In the framework of the generalized form of Fick's law, the molar flux of component i in a mixture of four components is written as a linear combination of concentration gradients ∇c_j [10]

$$J_i = -\sum_{j=1}^{3} D_{ij} \nabla c_j \,, \quad (i = 1, 2, 3) \,, \tag{3}$$

where D_{ii} are the main diffusion coefficients that relate the molar flux of component i to its own concentration gradient and D_{ij} are the cross diffusion coefficients that relate the molar flux of component i to the concentration gradient of component j. The Fick approach involves three independent diffusion fluxes and a 3×3 diffusion coefficient matrix, which is generally not symmetric, i.e. $D_{ij} \neq D_{ji}$. Further, the numerical values of D_{ij} depend both on the reference frame for velocity (molar-, mass- or volume-averaged) and on the order of the components.

The main shortcoming of Fick's law is the fact that concentration gradients are not the true thermodynamic driving forces for diffusion, which are rather given by chemical potential gradients. Maxwell-Stefan theory follows this path, assuming that chemical potential gradients $\nabla \mu_i$ are balanced by friction forces between the components that are proportional to their mutual velocity [55]. The Maxwell-Stefan diffusion coefficient $Đ_{ij}$ plays the role of an inverse friction coefficient between components i and j [55] and its matrix is symmetric so that it has only six independent elements. Maxwell-Stefan diffusion coefficients are associated with chemical potential gradients and thus cannot directly be measured in the laboratory. However,

they are accessible with equilibrium MD techniques, i.e. the Green-Kubo formalism or the Einstein approach.

This work employs the Green-Kubo formalism based on the net velocity autocorrelation function to obtain $n \times n$ phenomenological coefficients [37]

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

in a mixture of n components. Here, N is the total number of molecules, N_i is the number of molecules of component i and $\mathbf{v}_{i,k}(t)$ denotes the center of mass velocity vector of the k-th molecule of component i at time t.

Starting from the phenomenological coefficients L_{ij}, the elements of a $(n-1) \times (n-1)$ matrix $\boldsymbol{\Delta}$ can be defined as [37]

$$\Delta_{ij} = (1 - x_i) \left(\frac{L_{ij}}{x_j} - \frac{L_{in}}{x_n} \right) - x_i \sum_{k=1 \neq i}^{n} \left(\frac{L_{kj}}{x_j} - \frac{L_{kn}}{x_n} \right), \tag{5}$$

where x_i is the molar fraction of component i. Its inverse matrix $\mathbf{B} = \boldsymbol{\Delta}^{-1}$ is related to the Maxwell-Stefan diffusion coefficients $Đ_{ij}$.

On the other hand, experimental methods yield the Fick diffusion coefficients. Thus, to compare the predictions by molecular simulation with experimental values, a relation between Fick and Maxwell-Stefan diffusion coefficients is required [55]

$$\mathbf{D} = \boldsymbol{\Delta} \cdot \boldsymbol{\Gamma}, \tag{6}$$

in which all three symbols represent 3×3 matrices and the elements of $\boldsymbol{\Delta}$ are given by Eq. (5). $\boldsymbol{\Gamma}$ is the thermodynamic factor matrix

$$\Gamma_{ij} = \delta_{ij} + x_i \left. \frac{\partial \ln \gamma_i}{\partial x_j} \right|_{T,p,x_{k,k\neq j=1\ldots3}}. \tag{7}$$

Therein, δ_{ij} is the Kronecker delta function and γ_i the activity coefficient of component i. Here, the thermodynamic factor matrix was estimated from information on the microscopic structure given by radial distribution functions $g_{ij}(r)$ based on Kirkwood-Buff theory. In the grand canonical (μVT) ensemble, Kirkwood-Buff integrals G_{ij} are defined by [33]

$$G_{ij} = 4\pi \int_0^\infty \big(g_{ij}(r) - 1\big)\, r^2 dr. \tag{8}$$

Because the canonical (NVT) ensemble was employed, possible convergence issues [41] were corrected with the method by Krüger et al. [38]. Moreover, corrections of the radial distribution functions are required. Therefore, Kirkwood-Buff integrals were calculated based on the methodology proposed by Ganguly and van

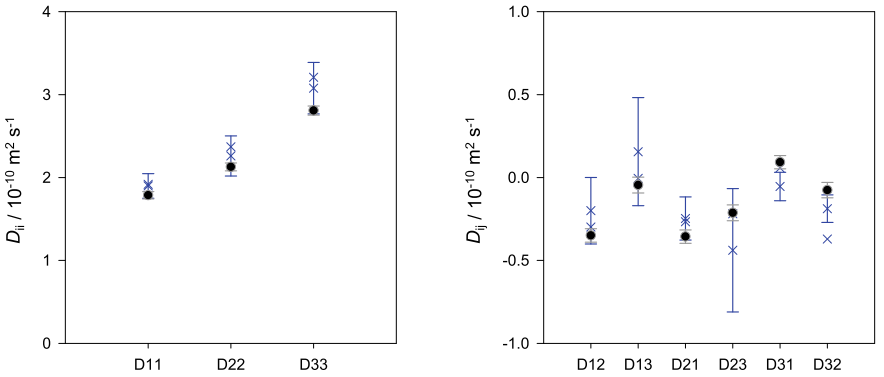


Fig. 5 Main (left) and cross (right) elements of the Fick diffusion coefficient matrix of the mixture cyclohexane (1) + toluene (2) + acetone (3) + methanol (4) at 298.15 K and 0.1 MPa. Present simulation results (black bullets) are compared with the experimental data [43] estimated from three and twelve experiments (blue crosses)

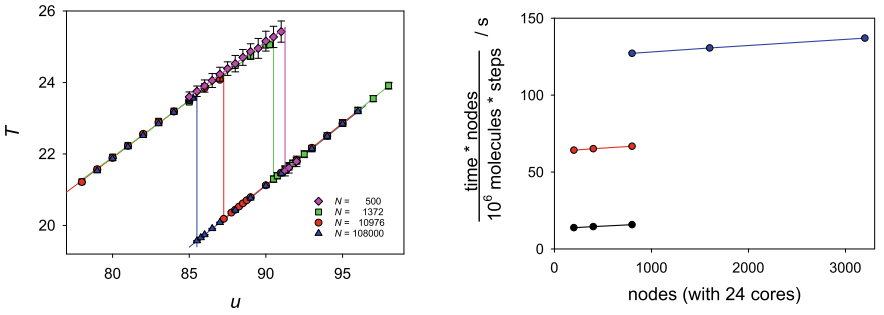
der Vegt [16]. Extrapolation to the thermodynamic limit was not necessary because of the rather large ensemble size $N = 8000$.

Predictive equilibrium MD simulations of diffusion coefficients and the thermodynamic factor of the quaternary mixture cyclohexane (1) + toluene (2) + acetone (3) + methanol (4) were carried out at 298.15 K and 0.1 MPa for one composition that was studied experimentally [43], i.e. $x_1 = x_2 = x_3 = 0.05$ mol mol^{-1}. A cubic simulation volume containing 8000 molecules with a cut-off radius of 24.5 Å was employed for this purpose. The resulting phenomenological coefficients L_{ij} were averaged from more than 10^5 correlation functions with a length of 20 ps. These coefficients were employed together with the thermodynamic factor matrix to calculate the Fick diffusion coefficient matrix in the molar frame of reference. In order to compare simulation results with the experimental data, the Fick diffusion coefficient matrix was transformed into the volume-averaged frame [55].

A comparison between present simulation results and experimental data is given in Fig. 5. The simulation results for all elements of the diffusion matrix agree with the experimental data within the reported uncertainties.

5 Solid/Fluid Phase Transition and Strong Scaling of *ms*2

In a recent study [40], the solid/fluid (S/F) phase transition was analyzed for the face centered cubic (fcc) lattice utilizing *ms*2 [44]. The LJ potential was applied in MD simulations such that the solid was heated at constant volume up to its phase transition. The Z method [5] was applied to determine the limit of superheating (LS) and the melting point (MP). For this purpose, total energy u (potential + kinetic energy) and temperature T were evaluated, cf. Fig. 6a. First, the fcc lattice remains



(a) (b)

Fig. 6 (a) Z method sampled by MD simulations using $ms2$; pink: $N = 500$; green: $N = 1372$; red: $N = 10976$; blue: $N = 108000$ atoms; vertical lines indicate the temperature drop from the LS to the MP. (b) Strong scaling efficiency of MD simulations with $ms2$ for a fcc LJ solid measured on CRAY XC40 (*Hazel Hen*) with hybrid MPI + OpenMP parallelization (each MPI process had two OpenMP threads); black: $N = 64000$ with a cutoff of $r_c = 6\sigma$; red: $N = 64000$ with $r_c = 29\sigma$; blue: $N = 120000$ with $r_c = 36\sigma$

for a specific total energy range in the metastable solid state, which is limited by u_{LS} when the solid melts. Beyond this range (u reaches values slightly above u_{LS}), the temperature drops to its melting temperature T_m since kinetic energy supplies the internal energy of fusion. At the S/F transition, the total energy $u^{solid}(v, T_{LS}) = u^{fluid}(v, T_m)$ holds with the maximum T_{LS} and minimum temperature T_m when a constant volume is maintained [40].

The melting process is strongly dependent on the system's structure and dynamics, particularly when a perfect fcc lattice without defects is considered. Thus, finite size effects were hypothesized and supported by a recent study [40]. Figure 6a clearly shows a substantial system size dependence of T_{LS} and T_m. Moreover, this behavior reinforces how well molecular simulations are able to tackle physical phenomena from a theoretical point of view, where experiments are challenging or even impossible because of extreme conditions.

In this context, the strong scaling efficiency of $ms2$ was analyzed for its hybrid MPI + OpenMP parallelization. Combining MPI and OpenMP, the memory demand of $ms2$ was optimized such that simulations with a larger particle number N can be achieved. In Fig. 6b, the vertical axis shows the computing power (nodes) times computing time per computing intensity (problem size), so that, horizontal lines would show a strong scaling efficiency of 100 %. From Fig. 6b, it becomes clear that $ms2$ is close to optimal strong scaling. However, the computing intensity of traversing the particle matrix is proportional to N^2, but intermolecular interactions are calculated for particles that are in the cutoff sphere only. As a result, the scaling of $ms2$ should be in between N and N^2. Figure 6b indicates that almost doubling the number of particles ($120000/64000 = 1.875$) in $ms2$ leads to an increase of computational cost of a factor around 1.91 if the number of nodes was chosen appropriately so that the overhead is small (comparison of red and blue symbols for 800 nodes).

6 Relative Permittivity of Mixtures

The relative permittivity of a fluid ε, also known as dielectric constant, indicates how that fluid weakens an external electric field compared to vacuum. While experimental data on the relative permittivity are available for many pure fluids (at least under ambient conditions), measurements of the relative permittivity for mixtures have rarely been reported. However, such information is important for chemical engineering, e.g. for electrolyte solutions with mixed solvents or solutions of weak electrolytes [35].

On the molecular scale, the relative permittivity is directly related to the mutual orientation of the molecular dipoles via Kirkwood's theory [32]. Thus, the relative permittivity can be sampled straightforwardly with molecular simulations in the canonical (NVT) ensemble via

$$\varepsilon - 1 = \frac{4\pi}{3k_\mathrm{B}TV}(\langle \boldsymbol{M}^2\rangle - \langle \boldsymbol{M}\rangle^2), \tag{9}$$

where k_B is Boltzmann's constant, T the temperature, V the volume and $\boldsymbol{M}$ the total dipole moment of the simulation volume that is obtained by summing up all molecular dipole moment vectors

$$\boldsymbol{M} = \sum_{i=1}^{N} \boldsymbol{\mu}_i. \tag{10}$$

Hence, molecular simulations are an ideal tool to study the relative permittivity of mixtures. It has recently been shown that with existing molecular models for mixtures of molecular fluids [34] and electrolyte solutions [45], at least qualitative agreement with experimental data can be obtained. To further demonstrate the power of this predictive approach also in a quantitative manner, MD simulations of the relative permittivity of the mixture acetone + water were carried out with the molecular simulation tool *ms*2 [44], using the TIP4P/ε water model [15] and the acetone model by Windmann et al. [58]. These models are known to yield the pure component permittivities excellently. The Lorentz-Berthelot combining rules were applied so that the simulation results for the mixture are strictly predictive. First, the mixture density was obtained with isothermal-isobaric (NpT) runs and then the relative permittivity was sampled with NVT simulations. The results in Fig. 7 demonstrate that a good prediction of the mixture permittivity was obtained.

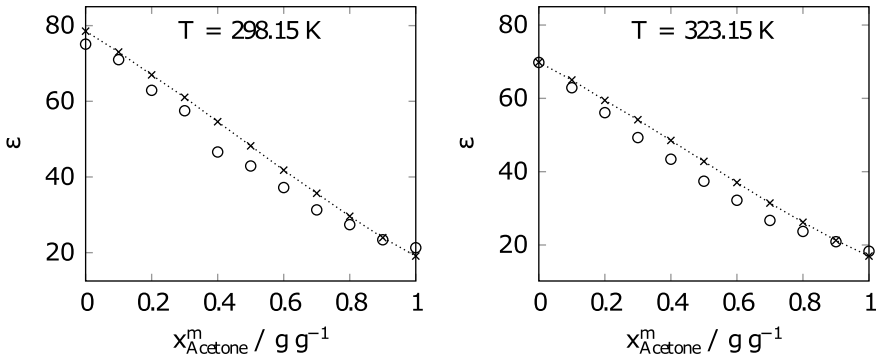


Fig. 7 Relative permittivity of mixtures of water and acetone as a function of the acetone mass fraction at two different temperatures and a pressure of 1 bar. Crosses show the experimental data by Åkerlöf [1], open circles denote present simulation results. Simulation uncertainties are within symbol size, dotted lines are guides to the eye

7 Reliability and Reproducibility of Simulation Data

Molecular simulations have become a well established alternative to laboratory experiments for predicting thermophysical properties of fluids [50, 51, 53]. Evidently, the reliability and reproducibility of such predictions is of fundamental importance.

To sample thermophysical properties of a given molecular model, computer experiments can be carried out. In general, the simulation result of a given observable x^{sim} will not agree with the true model value x^{mod} [20]. Like in laboratory experiments, errors can also occur in computer experiments [46] that can cause deviations between the true value x^{mod} and the value observed in simulation x^{sim} [20, 46]. Both stochastic and systematic errors may in general occur in computer experiments. While techniques to assess statistical errors are well established for computer simulations [2, 13, 14], it is more difficult to deal with systematic errors, which have a significant influence on the reliability of the results. Systematic errors may be a consequence of erroneous algorithms, user errors, differences due to different simulation methods, finite size effects, erroneous evaluation of long-range interactions, insufficient equilibration or production periods, compilers, parallelization, hardware architecture etc. [46]. As in laboratory experiments, round Robin studies can be made for quantifying systematic errors, in which the same simulation task is carried out by different groups with different programs.

The detection and assessment of outliers in large datasets is a standard task in the field of data science, but has to the best of our knowledge not yet been applied to thermophysical property data obtained by molecular simulation. The assessment of experimental thermophysical property data is a well-established field in chemical engineering [11], especially for phase equilibrium data [30, 31].

The accuracy with which properties of a simple (Lennard-Jones) model fluid can currently be determined by molecular simulation was assessed. The Lennard-Jones

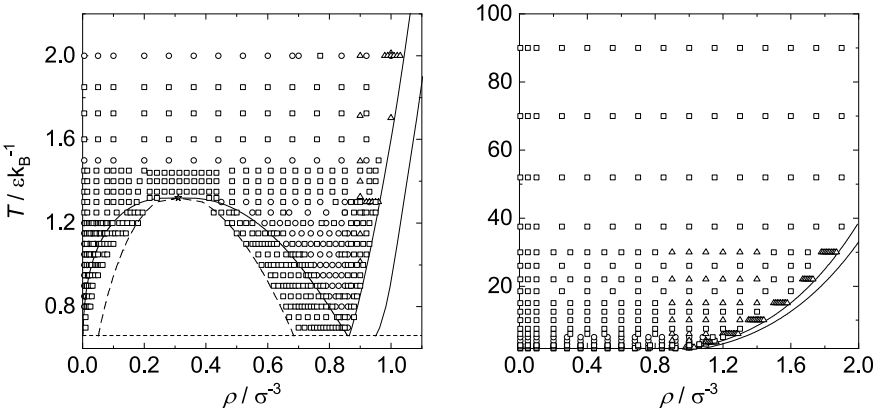


Fig. 8 Overview of about 1000 state points that were studied with the Lustig formalism [39] by different authors. Circles: Thol et al. [57]; triangles: Köster et al. [36]; squares: this work. Data for the Helmholtz energy and its density and inverse temperature derivatives up to second order were available for the stable state points. For the metastable state points, the derivatives were available up to first order

potential is often used as a starting point for the development of many force fields for complex molecules [52]. It is often taken as a benchmark for the validation of simulation codes and the test of new simulation techniques. Accordingly, a large number of computer experiment data are available for this fluid. Molecular simulations were performed both for homogeneous state points and for the vapor-liquid equilibrium to complement the data in regions that were only sparsely investigated in the literature. This database (cf. Fig. 8) allows for a systematic data evaluation and determination of outliers. In total, about 35,000 data points were evaluated [54]. The VLE properties: vapor pressure, saturated densities, enthalpy of vaporization and surface tension were investigated; for homogeneous state points, the investigated properties were: pressure, thermal expansion coefficient, isothermal compressibility, thermal pressure coefficient, internal energy, isochoric heat capacity, isobaric heat capacity, Grüneisen parameter, Joule-Thomson coefficient, speed of sound, Helmholtz energy and chemical potential.

Different consistency tests were applied to assess the accuracy and precision and thereby the reliability of the data. The data on homogeneous states were evaluated point-wise using data from their respective vicinity and EOS. Approximately 10% of all homogeneous bulk data were identified as gross outliers. VLE data were assessed by tests based on the compressibility factor, the Clausius-Clapeyron equation and by an outlier test. First, consistency tests were used to identify unreliable datasets. In a subsequent step, the mutual agreement of the remaining datasets was evaluated. Seven particularly reliable VLE data sets were identified. The mutual agreement of these data sets is approximately $\pm 1\%$ for vapor pressure, $\pm 0.2\%$ for saturated liquid density, $\pm 1\%$ for saturated vapor density and $\pm 0.75\%$ for enthalpy of vaporization – excluding the extended critical region. In most cases, the results from different

datasets were found to differ by more than the combined statistical uncertainty of the individual data. Hence, the magnitude of systematic errors often exceeds that from stochastic errors.

8 Data Management

While it is generally always advisable to follow good practices of data management when dealing with research data, this becomes even more expedient in cases where the data have been obtained by accessing dedicated facilities, as it is the case in scientific high-performance computing: Simulation results without annotation become *dark data*, making their meaning and purpose unintelligible to others, in particular, to automated processing on repositories and computing environments [47]. HPC and other facilities are not employed adequately if they are used to generate dark data. Obversely, annotating the simulation outcome with appropriate metadata enhances its value to the community and ensures that data become and remain FAIR, i.e., findable, accessible, interoperable, and reusable, permitting their preservation far beyond the immediate circumstances that motivated their creation originally [6, 7, 48].

Data infrastructures, such as repositories, digital marketplaces or modelling and simulation environments, often follow a multi-tier design with an explicit logical or semantic layer, as illustrated by Fig. 9. In these cases, the underlying semantic technology, which may include non-relational databases, mechanisms for checking constraints, or handling digital objects, requires mechanisms for knowledge representation. This technical requirement is the main underlying cause of the increasing pressure on scientific communities to develop standardized schema metadata

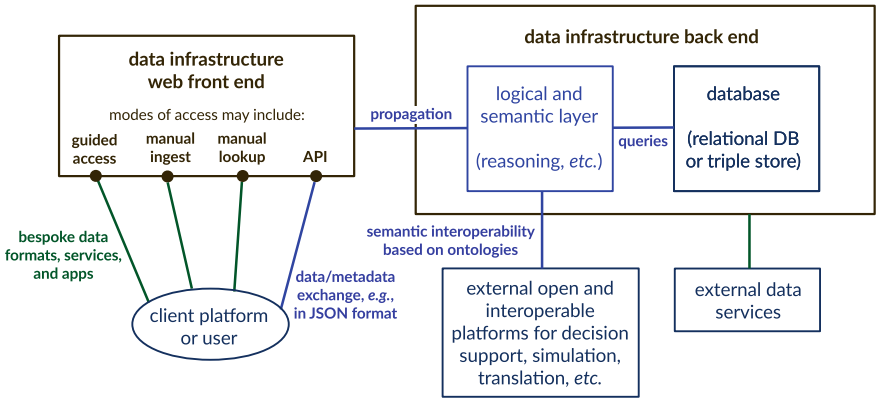


Fig. 9 Role of semantic technology within interoperable data infrastructures, illustrated for the case of a typical multi-tier architecture

definitions or ontologies. Such metadata standards are known as semantic assets; an agreement on semantic assets establishes semantic interoperability.

In materials modelling, understood here as roughly comprising the fields of computational molecular engineering (CME) for soft matter [29] and integrated computational materials engineering (ICME) for solid materials [49], a major community-governed effort towards metadata standardization is conducted by the European Materials Modelling Council (EMMC), specifically the EMMC focus areas on digitalization and interoperability, supported by a series of projects funded from the Horizon 2020 research and innovation programme, including VIMMP [28]. This approach, which is implemented by the present data management concept, is based on a system of ontologies that permits the characterization of CME/ICME data provenance at multiple levels of abstraction:

To facilitate *technical-level reproducibility*, metadata documenting all boundary conditions, technical parameters of the employed software and circumstances related to workflow execution need to be provided, including details on the hardware architecture and the mode of parallelization. Semantic assets that can be used for this purpose include the VIMMP ontologies MACRO, OSMO, VISO and VOV [28] in combination with the PaaSPort ontology [4]. Documenting a workflow manually in this way, at full detail, is not recommended except in the case of very straightforward scenarios; the complete viability of such an approach would require the automated annotation by an integration of ontologies with workflow management systems [29].

In a *logical representation* of a simulation workflow, details of the technical implementation are left out of consideration; instead, the workflow is described in terms of the involved use cases, models, solvers and processors, which are defined by their function rather than by their practical realization. This approach was introduced by the EMMC through the development of the MODA (Model Data) workflow description standard [8]; this is an adequate level of annotation for most purposes, as it permits documenting the provenance of simulation results as well as the intended use cases in a similar way as it is usually done in a scientific journal article. However, metadata provided in this way are machine-processable and standardized by an ontology – in the present case, by the Ontology for Simulation, Modelling, and Optimization (OSMO), i.e., the ontology version of MODA [29]. Logical data transfer (LDT) notation is a graph-based visualization of such workflow descriptions, which was also developed on the basis of a similar notation from MODA. The LDT graph corresponding to the scenario from Sect. 1 is shown in Fig. 10; this graph corresponds to a workflow description in terms of OSMO and to a collection of digital objects that can be ingested into (and extracted from) an interoperable data infrastructure, e.g., in JSON or JSON-LD format, cf. Fig. 9.

For a high-level representation of CME/ICME scenarios, a conceptualization of *modelling and simulation workflows as semioses* is developed on the basis of the European Materials and Modelling Ontology (EMMO) [12, 17]. As a top-level ontology, the main purpose of the EMMO consists in establishing the foundations for a coherent architecture of semantic assets at the highest possible degree of abstraction. Due to the nature of this work, technical and philosophical requirements need to be reconciled; the present stage of these developments is discussed in a recent report [27].

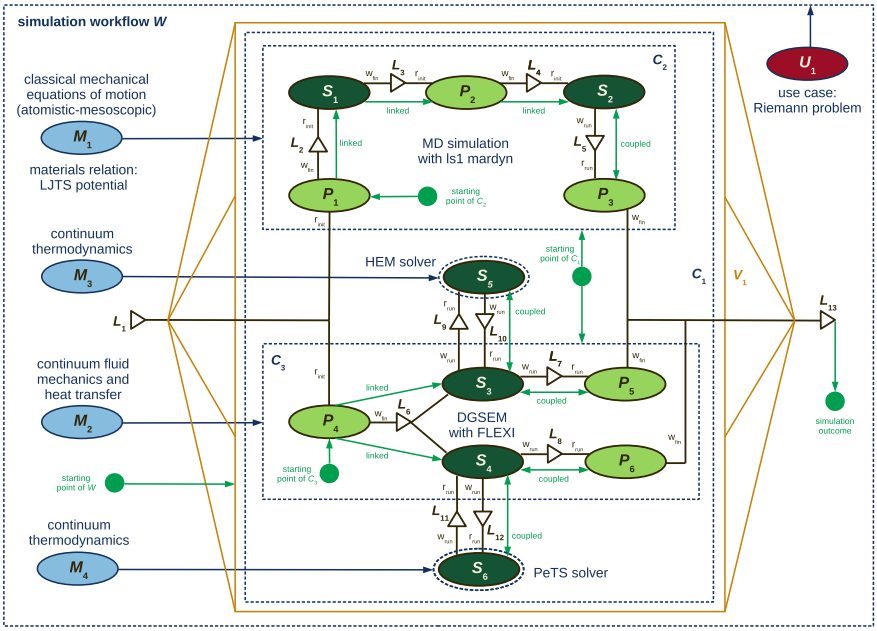


Fig. 10 Logical data transfer (LDT) provenance documentation for simulations by Hitz et al. [26] addressing the Riemann problem use case from Sect. 1. The LDT notation was presented in detail in previous work [29]

Acknowledgements The co-authors M.H., R.S.C., S.H., G.G.-C., R.F., M.K., S.S. and J.V. acknowledge funding by Deutsche Forschungsgemeinschaft (DFG) through the Project SFB-TRR 75, Project number 84292822 - "Droplet Dynamics under Extreme Ambient Conditions", and the co-author M.T.H. acknowledges funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no. 760907 (VIMMP). This work was carried out under the auspices of the Boltzmann-Zuse Society of Computational Molecular Engineering (BZS), and it was facilitated by activities of the Innovation Centre for Process Data Technology (Inprodat e.V.), Kaiserslautern. The simulations were performed on the CRAY XC40 (*Hazel Hen*) at the High Performance Computing Center Stuttgart (HLRS). Discussions with S. Chiacchiera, B. Andreon, E. Bayro Kaiser, W. L. Cavalcanti, A. Fiseni, G. Goldbeck, A. Scotto di Minico, M. A. Seaton, and I. T. Todorov are acknowledged.

References

1. G. Åkerlöf, Dielectric constants of some organic solvent-water mixtures at various temperatures. J. Am. Chem. Soc. **54**, 4125–4139 (1932)
2. M.P. Allen, D.J. Tildesley, *Computer Simulation of Liquids* (Oxford University Press, Oxford, 1989)
3. T. Allie-Ebrahim, V. Russo, O. Ortona, L. Paduano, R. Tesser, M. Di Serio, P. Singh, Q. Zhu, G.D. Moggridge, C. D'Agostino, A predictive model for the diffusion of a highly non-ideal ternary system. Phys. Chem. Chem. Phys. **20**, 18436–18446 (2018)

4. N. Bassiliades, M. Symeonidis, P. Gouvas, E. Kontopoulos, G. Meditskos, I. Vlhavas, PaaSPort semantic model: an ontology for a platform-as-a-service semantically interoperable marketplace. Data Knowl. Eng. **113**, 81–115 (2018)
5. A.B. Belonoshko, S. Davis, N.V. Skorodumova, P.H. Lundow, A. Rosengren, B. Johansson, Properties of the fcc Lennard-Jones crystal model at the limit of superheating. Phys. Rev. B **76**, 064121 (2007)
6. J. Bicarregui, Building and sustaining data infrastructures: putting policy into practice Technical report, Wellcome Trust (2016)
7. J. Bicarregui, N. Gray, R. Henderson, R. Jones, S. Lambert, B. Matthews, Data management and preservation planning for Big Science. Int. J. Digit. Curation **8**, 29–41 (2013)
8. CEN-CENELEC Management Centre, *Materials Modelling: Terminology, Classification and Metadata* (CEN workshop agreement, Brussels, 2018)
9. R.S. Chatwell, J. Vrabec, Bulk viscosity of liquid noble gases. J. Chem. Phys. **152**, 094503 (2020)
10. E.L. Cussler, *Mass Transfer in Fluid Systems*, 2nd edn. (Cambridge University Press, Cambridge, 1997)
11. Q. Dong, X. Yan, R.C. Wilhoit, X. Hong, R.D. Chirico, V.V. Diky, M. Frenkel, Data quality assurance for thermophysical property databases: applications to the TRC SOURCE data system. J. Chem. Inf. Comput. Sci. **42**, 473–480 (2002)
12. EMMC Coordination and Support Action: European Materials and Modelling Ontology (EMMO). (2020). https://github.com/emmo-repo/ and https://emmc.info/emmo-info/
13. H. Flyvbjerg, H.G. Petersen, Error estimates on averages of correlated data. J. Chem. Phys. **91**, 461–466 (1989)
14. D. Frenkel, Simulations: The dark side. Eur. Phys. J. Plus **128**, 10 (2013)
15. R. Fuentes-Azcatl, J. Alejandre, Non-polarizable force field of water based on the dielectric constant: TIP4P/ε. J. Phys. Chem. B **118**, 1263–1272 (2014)
16. P. Ganguly, N.F.A. van der Vegt, Convergence of sampling Kirkwood-Buff integrals of aqueous solutions with molecular dynamics simulations. J. Chem. Theory Comput. **9**, 1347–1355 (2013)
17. G. Goldbeck, E. Ghedini, A. Hashibon, G.J. Schmitz, J. Friis, A reference language and ontology for materials modelling and interoperability, in *Proceedings of the NAFEMS World Congress* 2019, Québec, (NAFEMS, Knutsford, UK, 2019), p. NWC_19_86
18. S. Grottel, M. Krone, C. Müller, G. Reina, T. Ertl, MegaMol–a prototyping framework for particle-based visualization. IEEE Trans. Vis. Comput. Graph. **21**, 201–214 (2015)
19. G. Guevara-Carrion, R. Fingerhut, J. Vrabec, Fick diffusion coefficient matrix of a quaternary liquid mixture by molecular dynamics. J. Phys. Chem. B **124**, 4527–4535 (2020)
20. H. Hasse, J. Lenhard, Boon and bane: on the role of adjustable parameters in simulation models. in *Mathematics as a Tool. Boston Studies in the Philosophy and History of Science*, vol. 327 ed. by J. Lenhard, M. Carrier, (Springer, Cham, 2017)
21. M. Heier, S. Stephan, J. Liu, W.G. Chapman, H. Hasse, K. Langenbach, Equation of state for the Lennard-Jones truncated and shifted fluid with a cut-off radius of 2.5 based on perturbation theory and its applications to interfacial thermodynamics. Mol. Phys. **116**, 2083–2094 (2018)
22. M. Heinen, J. Vrabec, Evaporation sampled by stationary molecular dynamics simulation. J. Chem. Phys. **151**, 044704 (2019)
23. M. Heinen, J. Vrabec, J. Fischer, Communication: Evaporation: Influence of heat transport in the liquid on the interface temperature and the particle flux. J. Chem. Phys. **145**, 081101 (2016)
24. F. Hindenlang, G.J. Gassner, C. Altmann, A. Beck, M. Staudenmaier, C.D. Munz, Explicit discontinuous Galerkin methods for unsteady problems. Comput. Fluids **61**, 86–93 (2012)
25. T. Hitz, M. Heinen, J. Vrabec, C.D. Munz, Comparison of macro- and microscopic solutions of the Riemann problem I. Supercritical shock tube and expansion into vacuum. J. Comput. Phys. **402**, 109077 (2020)
26. T. Hitz, S. Jöns, M. Heinen, J. Vrabec, C.D. Munz, Comparison of macro- and microscopic solutions of the Riemann problem II. Two-phase shock tube, J. Comput. Phys. **429**, 110027 (2021)

27. M.T. Horsch, S. Chiacchiera, M.A. Seaton, I.T. Todorov, *Multiscale Modelling and Simulation of Physical Systems as Semiosis* Technical report, Innovation Centre for Process Data Technology, Kaiserslautern (2020)
28. M.T. Horsch, S. Chiacchiera, M.A. Seaton, I.T. Todorov, K. Šindelka, M. Lísal, B. Andreon, E.B. Kaiser, G. Mogni, G. Goldbeck, R. Kunze, G. Summer, A. Fiseni, H. Brüning, P. Schiffels, W.L. Cavalcanti, Ontologies for the Virtual Materials Marketplace. KI - Künstliche Intelligenz **34**, 423–428 (2020)
29. M.T. Horsch, C. Niethammer, G. Boccardo, P. Carbone, S. Chiacchiera, M. Chiricotto, J.D. Elliott, V. Lobaskin, P. Neumann, P. Schiffels, M.A. Seaton, I.T. Todorov, J. Vrabec, W.L. Cavalcanti, Semantic interoperability and characterization of data provenance in computational molecular engineering. J. Chem. Eng. Data **65**, 1313–1329 (2020)
30. J.W. Kang, V. Diky, R.D. Chirico, J.W. Magee, C.D. Muzny, I. Abdulagatov, A.F. Kazakov, M. Frenkel, Quality assessment algorithm for vapor-liquid equilibrium data. J. Chem. Eng. Data **55**, 3631–3640 (2010)
31. J.W. Kang, V. Diky, R.D. Chirico, J.W. Magee, C.D. Muzny, A.F. Kazakov, K. Kroenlein, M. Frenkel, Algorithmic framework for quality assessment of phase equilibrium data. J. Chem. Eng. Data **59**, 2283–2293 (2014)
32. J. Kirkwood, The dielectric polarization of polar liquids. J. Chem. Phys. **7**, 911–919 (1939)
33. J.G. Kirkwood, F.P. Buff, The statistical mechanical theory of solutions. I. J. Chem. Phys. **19**, 774–777 (1951)
34. M. Kohns, Molecular simulation study of dielectric constants of pure fluids and mixtures. Fluid Phase Equilib. **506**, 112393 (2020)
35. M. Kohns, G. Lazarou, S. Kournopoulos, E. Forte, F.A. Perdomo, G. Jackson, C.S. Adjiman, A. Galindo, Predictive models for the phase behaviour and solution properties of weak electrolytes: nitric, sulphuric, and carbonic acids. Phys. Chem. Chem. Phys. **22**, 15248–15269 (2020)
36. A. Köster, P. Mausbach, J. Vrabec, Premelting, solid-fluid equilibria, and thermodynamic properties in the high density region based on the Lennard-Jones potential. J. Chem. Phys. **147**, 144502 (2017)
37. R. Krishna, J.M. van Baten, The Darken relation for multicomponent diffusion in liquid mixtures of linear alkanes: An investigation using Molecular Dynamics (MD) simulations. Ind. Eng. Chem. Res. **44**, 6939–6847 (2005)
38. P. Krüger, T.J.H. Vlugt, Size and shape dependence of finite-volume Kirkwood-Buff integrals. Phys. Rev. E **97**, 051301 (2018)
39. R. Lustig, Statistical analogues for fundamental equation of state derivatives. Mol. Phys. **110**, 3041–3052 (2012)
40. P. Mausbach, R. Fingerhut, J. Vrabec, Structure and dynamics of the Lennard-Jones fcc-solid focusing on melting precursors. J. Chem. Phys. **153**, 104506 (2020)
41. J. Milzetti, D. Nayar, N.F.A. van der Vegt, Convergence of Kirkwood-Buff integrals of ideal and nonideal aqueous solutions using molecular dynamics simulations. J. Phys. Chem. B **122**, 5515–5526 (2018)
42. 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, ls1 mardyn: The massively parallel molecular dynamics code for large systems. J. Chem. Theory Comput. **10**, 4455–4464 (2014)
43. C. Peters, J. Thien, L. Wolff, H.J. Koß, A. Bardow, Quaternary diffusion coefficients in liquids from microfluidics and raman microspectroscopy: cyclohexane + toluene + acetone + methanol. J. Chem. Eng. Data **65**, 1273–1288 (2020)
44. G. Rutkai, A. Köster, G. Guevara-Carrion, T. Janzen, M. Schappals, C.W. Glass, M. Bernreuther, A. Wafai, S. Stephan, M. Kohns, S. Reiser, S. Deublein, M. Horsch, H. Hasse, J. Vrabec, ms2: A molecular simulation tool for thermodynamic properties, release 3.0. Comput. Phys. Commun. **221**, 343–351 (2017)

45. D. Saric, M. Kohns, J. Vrabec, Dielectric constant and density of aqueous alkali halide solutions by molecular dynamics: a force field assessment. J. Chem. Phys. **152**, 164502 (2020)
46. M. Schappals, A. Mecklenfeld, L. Kröger, V. Botan, A. Köster, S. Stephan, E.J. Garcia, G. Rutkai, G. Raabe, P. Klein, K. Leonhard, C.W. Glass, J. Lenhard, J. Vrabec, H. Hasse, Round Robin study: molecular simulation of thermodynamic properties from models with internal degrees of freedom. J. Chem. Theory Comput. **13**, 4270–4280 (2017)
47. B. Schembera, J.M. Durán, Dark data as the new challenge for big data science and the introduction of the scientific data officer. Philos. Tech. **33**, 93–115 (2019)
48. B. Schembera, D. Iglezakis, EngMeta: metadata for computational engineering. Int. J. Metadata Semant. Ontol. **14**, 26 (2020)
49. G.J. Schmitz, Microstructure modeling in integrated computational materials engineering (ICME) settings: can HDF5 provide the basis for an emerging standard for describing microstructures? JOM **68**, 77–83 (2015)
50. S. Stephan, S. Becker, K. Langenbach, H. Hasse, Vapor-liquid interfacial properties of the binary system cyclohexane + CO2: Experiment, molecular simulation and density gradient theory. Fluid Phase Equilib. **518**, 112583 (2020)
51. S. Stephan, H. Hasse, Interfacial properties of binary mixtures of simple fluids and their relation to the phase diagram. Phys. Chem. Chem. Phys. **22**, 12544–12564 (2020)
52. S. Stephan, M. Horsch, J. Vrabec, H. Hasse, MolMod - an open access database of force fields for molecular simulations of fluids. Mol. Simul. **45**, 806–814 (2019)
53. S. Stephan, K. Langenbach, H. Hasse, Interfacial properties of binary Lennard-Jones mixtures by molecular simulations and density gradient theory. J. Chem. Phys. **150**, 174704 (2019)
54. S. Stephan, M. Thol, J. Vrabec, H. Hasse, Thermophysical properties of the Lennard-Jones fluid: database and data assessment. J. Chem. Inf. Model. **59**, 4248–4265 (2019)
55. R. Taylor, R. Krishna, *Multicomponent Mass Transfer* (John Wiley & Sons, New York, 1993)
56. N. Tchipev, S. Seckler, M. Heinen, J. Vrabec, F. Gratl, M. Horsch, M. Bernreuther, C.W. Glass, C. Niethammer, N. Hammer, B. Krischok, M. Resch, D. Kranzlmüller, H. Hasse, H.J. Bungartz, P. Neumann, Twetris: Twenty trillion-atom simulation. Int. J. High Perform. Comput. Appl. **33**, 838–854 (2019)
57. M. Thol, G. Rutkai, A. Köster, R. Lustig, R. Span, J. Vrabec, Equation of state for the Lennard-Jones fluid. J. Phys. Chem. Ref. Data **45**, 023101 (2016)
58. T. Windmann, M. Linnemann, J. Vrabec, Fluid phase behavior of nitrogen + acetone and oxygen + acetone by molecular simulation, experiment and the Peng-Robinson equation of state. J. Chem. Eng. Data **59**, 28–38 (2014)

Phase-Field Simulations with the Grand Potential Approach

M. Seiz, P. Hoffrogge, H. Hierl, A. Reiter, D. Schneider, and B. Nestler

Abstract Simulations have become a powerful tool for understanding physical processes and help the development of new products. Many of these new products rely on controlling the properties of their materials precisely. These properties are closely linked to the inner microstructure of the material, which the phase-field method can predict. Based on a grand potential approach, several results from different processes will be shown. Additionally, optimizations for one kind of process will be discussed and their effect on the performance and scalability shown.

Keywords Phase-field · Simulation · Vectorization · Parallel scaling · Freeze-casting · Sintering · SOFC

1 Introduction

The properties of materials depend on both the chemical composition and on the microstructure, which is dependent on the manufacturing process and its parameters. The phase-field method is an appealing technique to study various processes in materials science and engineering where complex microstructural phenomena play an important role. Take the sintering process as an example: A loosely bound powder (green body) is heated up to a significant fraction of its melting point. The individual particles of the green body begin to diffuse and approach each other to minimize the total energy. Tracking the changes during this process with experiments

M. Seiz (✉) · P. Hoffrogge · H. Hierl · D. Schneider · B. Nestler
Institute of Applied Materials (IAM), Karlsruhe Institute of Technology (KIT),
Straße am Forum 7, 76131 Karlsruhe, Germany
e-mail: marco.seiz@kit.edu

A. Reiter · D. Schneider · B. Nestler
Institute of Digital Materials, Hochschule Karlsruhe Technik und Wirtschaft,
Moltkestr. 30, 76131 Karlsruhe, Germany

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_37

requires in-situ time-resolved 3D computer tomography, which is a quite involved and expensive process. In contrast to this, the phase-field method naturally allows tracking the changes during a simulation run. While a model includes many simplifications compared to reality, the phase-field model delivers three-dimensional data about the temporal evolution of microstructure which remains in many applications challenging, if not infeasible, to obtain in an experimental setup.

This report focuses predominantly on the application of multi phase-field models in the field of high performance computing on modern supercomputers. Implementation details are discussed in addition to few simulation examples. Within the manuscript, the following applications of the phase-field models are presented:

- Stability of planar fronts during the freeze-casting process (ForHLR II)
- Performance optimizations for a 3-component phase-field model for nickel coarsening in a Ni-YSZ SOFC-anode (ForHLR II)
- Scalability and time-to-solution of an optimized 2-component alumina sintering model for different time integration schemes (Hazel Hen)

First, the employed model is described with the differences between the simulations also being discussed. Next, the implementation of the models within the massively parallel framework PACE3D is detailed. Based on this the individual results are presented and conclusions are drawn.

2 Model

The phase-field model of [1, 9, 19] based on the grand potential approach is used. The N different phases are described by the order-parameters ϕ_α, with $\alpha \in \{1, \ldots, N\}$ [10]. Each ϕ_α can be interpreted as the local volume fraction of a phase α, each of which may differ in its state of matter, crystallographic arrangement and orientation. These order-parameters are collected in the phase-field vector ϕ. The K concentrations c_i and their corresponding chemical potentials μ_i of the components $i \in \{1, \ldots, K\}$ are described by the concentration vector $\mathbf{c}$ and chemical potential vector μ, respectively [2, 19].

The evolution equation for ϕ reads

$$\begin{aligned} \tau\epsilon\frac{\partial\phi_\alpha}{\partial t} \quad &= \quad -\epsilon\left(\frac{\partial a(\phi,\nabla\phi)}{\partial\phi_\alpha} - \nabla\cdot\frac{\partial a(\phi,\nabla\phi)}{\partial\nabla\phi_\alpha}\right) - \frac{1}{\epsilon}\frac{\partial w(\phi)}{\partial\phi_\alpha} \\ &\quad -\sum_{\beta=1}^{N}\psi_\beta(\mu,T)\frac{\partial h_\beta(\phi)}{\partial\phi_\alpha} - \lambda \end{aligned} \tag{1}$$

with the kinetic coefficient τ, the gradient energy a, the obstacle potential w, the grand potential ψ_β and a Lagrange multiplier λ. The kinetic coefficient τ controls the timescale on which ϕ evolves and is calculated as

$$\tau = \left(\sum_{\substack{\alpha,\beta=1 \\ (\alpha<\beta)}}^{N,N} a_{kin}(\phi, \nabla\phi)\tau_{\alpha\beta}\phi_\alpha\phi_\beta \right) \left(\sum_{\substack{\alpha,\beta=1 \\ (\alpha<\beta)}}^{N,N} \phi_\alpha\phi_\beta \right)^{-1} \tag{2}$$

with the pairwise $\tau_{\alpha\beta}$ coefficients for each interface and a kinetic anisotropy term a_{kin}. The gradient energy a and the obstacle potential w control the equilibrium interface shape and integrate the effects of surface energy into the phase-field model with the pairwise surface energies $\gamma_{\alpha\beta}$. The interface is about 2.5ϵ thick. The grand potentials ψ_β are the driving forces for non-capillary interface motion and are interpolated with the h_β of [16]. The Lagrange multiplier $\lambda = \frac{1}{N}\sum_{\beta=1}^{N} rhs_\beta$ accounts for the constraint $\sum_{\alpha=1}^{N} {}^{\partial\phi_\alpha}/_{\partial t} = 0$. Note that in the case of SOFC-modeling, a slightly different model formulation following Steinbach and Pezzolla [23] is employed, which obviates both the use of a Lagrange multiplier and interpolation of the kinetic coefficient according to Eq. (2).

For the three presented results, the main difference for Eq. (1) lies in the number of phases N. In case of the solidification of a planar front, only two phases are required. For SOFC degradation, three distinct phases are necessary, but as the mechanism is surface diffusion controlled, the grain boundaries need not be resolved. Hence three order parameters are employed. In contrast to these, sintering requires the resolution of individual particles (grains) and their grain boundaries, hence many phases are necessary. This would ordinarily lead to at least a linear dependence of the memory and computational requirements on the number of phases, as Eq. (1) needs to be solved for each phase ϕ_α. By implementing a locally reduced order parameter (LROP) approach [14], the memory and computational effort are kept independent of the number of phases. In the LROP approach, only a fixed number of phases are saved per voxel cell, with the results of [14] suggesting that 6 local phases are sufficient to reproduce the correct behavior. Due to vectorization over phases, the sintering model uses 8 local phases.

Besides the number of phases, the gradient energy a and the kinetic coefficient τ are anisotropic in case of solidification. The anisotropy is introduced by multiplying the interpolated gradient energy term respectively kinetic coefficient with an appropriate anisotropy function as shown in Eq. (2) with $a_{kin}(\phi, \nabla\phi)$. The anisotropy function needs to account for the phases present as well as the interface orientation and hence depends on both ϕ and $\nabla\phi$.

The phase-field equation is coupled to the evolution of the chemical potential

$$\begin{aligned} \partial\mu = \left[\sum_{\alpha=1}^{N} h_\alpha(\phi) \left(\frac{\partial \mathbf{c}_\alpha(\mu, T)}{\partial\mu} \right) \right]^{-1} \Bigg(\nabla \cdot \Big(\mathbf{M}(\phi, \mu, T)\nabla\mu - \mathbf{J}_{at}(\phi, \mu, T) \Big) \\ - \sum_{\alpha=1}^{N} \mathbf{c}_\alpha(\mu, T) \frac{\partial h_\alpha(\phi)}{\partial t} - \sum_{\alpha=1}^{N} h_\alpha(\phi) \left(\frac{\partial \mathbf{c}_\alpha(\mu, T)}{\partial T} \right) \partial T \Bigg), \end{aligned} \tag{3}$$

which is related to the conservation of mass via the conserved concentration field $\mathbf{c}$. The change of concentration due to fluxes and coupling to the phase-field and temperature are calculated and then transformed into a change of chemical potential with the generalized susceptibility $\frac{\partial \mu}{\partial c}$. The phase concentrations $\mathbf{c}_\alpha$ are calculated from the chemical potential μ and the temperature T. The anti-trapping current J_{at} prevents effects of the artificially enlarged interface width in case of solidification [1, 3, 12] and does not enter the other models. The other two terms within the brackets are the changes of concentration due to the coupling the chemical potential to the phase-field and the temperature.

The mobility $\mathbf{M}$ is related to the diffusivity $\mathbf{D}_\alpha$ which is a vector of K components per phase α. In case of solidification, only volume diffusion is considered and hence $\mathbf{M} = \sum_{\alpha=1}^{N} h_\alpha^\mu(\phi)\mathbf{M}_\alpha$ with $\mathbf{M}_\alpha = \mathbf{D}_\alpha \partial \mathbf{c}_\alpha(\mu,T)/\partial\mu$ being the chemical mobility within phase α for the K components. The function $h_\alpha^\mu(\phi) = \phi_\alpha$ is used for interpolation for computational efficiency [13, 20]. In contrast to this, SOFC-degradation and sintering requires the inclusion of boundary diffusion, which is incorporated as in [8] yielding

$$\mathbf{M} = \sum_{\alpha=1}^{N} h_\alpha^\mu(\phi)\mathbf{M}_\alpha \tag{4}$$

$$+ \sum_{\substack{\alpha=1 \\ \alpha\neq v}}^{N} g(\phi_\alpha, \phi_v)\mathbf{D}_s \left(\frac{\partial \mathbf{c}_\alpha(\mu,T)}{\partial\mu}\phi_\alpha + \frac{\partial \mathbf{c}_v(\mu,T)}{\partial\mu}\phi_v \right) \tag{5}$$

$$+ \sum_{\substack{\alpha=1 \\ \alpha\neq v}}^{N} \sum_{\substack{\beta=1 \\ \beta\neq\alpha \\ \beta\neq v}}^{N} g(\phi_\alpha, \phi_\beta)\mathbf{D}_{gb} \left(\frac{\partial \mathbf{c}_\alpha(\mu,T)}{\partial\mu}\phi_\alpha + \frac{\partial \mathbf{c}_\beta(\mu,T)}{\partial\mu}\phi_\beta \right) \tag{6}$$

with the surface diffusivity $\mathbf{D}_s$, the grain boundary diffusivity $\mathbf{D}_{gb}$ and the index of the surrounding vapor phase v. The function $g(\phi_\alpha, \phi_\beta) = \phi_\alpha\phi_\beta$ identifies the regions where these boundary diffusion mechanisms are active.

For the SOFC degradation, the three components Ni, YSZ and Pore ($K = 3$) are introduced while the three components are treated as near-immiscible by simple parabolic free-energies whereas for the sintering of alumina the two components Al and O ($K = 2$) are employed. In the SOFC-model, surface diffusion is assumed to be the dominant mass-transfer mechanism for nickel, while the YSZ-component in the SOFC-model is treated as stable by assigning vanishing diffusivities to it.

Finally, in the case of solidification an external temperature gradient is applied by

$$T(x,t) = T_0 + G(x - vt)$$

with the gradient G, its velocity v and x pointing in the growth direction. The SOFC degradation and sintering models are assumed to be isothermal and hence the temperature dependence in the equations thus far vanishes.

A paper is under preparation for the parametrization and further results on freeze-casting process. Further specifics including the concrete parametrization of the alumina sintering model can be found in [8]. While the current manuscript focuses on the implementation and optimization of the phase-field model at hand, details of the parametrization and modelling approach of nickel-coarsening in SOFC-anodes are omitted here for the sake of brevity. These details are planned to be included in at least one future publication.

3 The PACE3D framework

The phase-field models are implemented in the massively parallel framework PACE3D [7]. A wide range of multi-physics models is available within the solver in order to study the microstructure evolution for many processes. Parallelization is achieved by spatial domain decomposition using the message passing interface (MPI). Each rectangular subdomain is assigned a rank and extended by ghost layers. The ghost layer exchange can be done with both blocking and non-blocking send and receive functions. If diagonal cells are required by the stencil, the send and receive functions are executed separately for each dimension to avoid diagonal exchanges between the subdomains. The solver scales almost ideally on up to 172 032 cores on different German supercomputers like the Hazel Hen, the SuperMUC-NG as well as the ForHLR I and II. On an Intel Xeon CPU E5-2680 v3 a single-core peak performance of 13 GFLOP/s was reached, corresponding to 32.5 % of the theoretical peak performance [6]. MPI I/O is used for storing and reading the voxel fields. Additionally, the voxel fields can be turned into surface meshes employing a marching cubes algorithm. This generally reduces the size, which allows higher output rates for visualization.

4 Results

4.1 Stability During Freeze-Casting

Freeze-casting is a novel manufacturing technique with applications including filters [15], near-net shape casting [22] and biomaterials [21, 26] e.g. for bone replacement. A suspension is brought below its freezing point with the fluid solidifying. Depending on the processing parameters and the fluid plus its additives, many different microstructures may be achieved. But for any application relevant microstructure

(a) initial conditions (b) example of the onset of instability

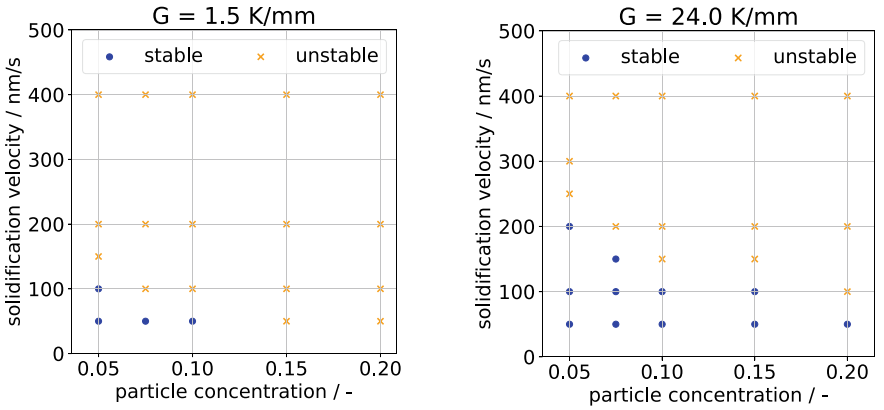
Fig. 1 Initial conditions of the simulation as well an exemplary unstable simulation. Yellow indicates the ice crystal and blue the suspension, with the red-orange part being the diffuse interface

to appear, the initially flat interface has to destabilize. Hence the stability of a planar ice front against an undercooled suspension is investigated.

The simulations are conducted as follows: A planar ice front is put in one part of the domain and the rest of the domain is filled with the suspension at the investigated concentration c_0. In the immediate vicinity of the solid front, a higher concentration of particles is assumed, corresponding to 0.9 times the equilibrium concentration of the suspension—this reduces the initial driving force which helps avoid stretched interfaces. Periodic boundary conditions are applied on the sides, whereas zero flux conditions are applied on the solid side. On the liquid side of the domain the phase-field has zero flux conditions and the concentration is fixed to the concentration c_0 with a Dirichlet boundary condition. A moving window technique is applied in order to simulate a quasi-infinite domain in the growth direction. The planar ice front is set such that this moving window starts immediately. A graphical overview of the simulation setup can be seen in Fig. 1a and a visibly perturbed state Fig. 1b.

All simulations were ran on 20 cores (1 node) of the ForHLR II at the SCC Karlsruhe.

Based on experimental results [25] the velocity range 50 nm/s to 400 nm/s and the concentration range 0.05 to 0.2 are chosen for this study. Two temperature gradients are investigated, 1.5 K/mm which is close to the experiments and 24 K/mm in order to observe the effects of an increased gradient. The simulations were run until the interface destabilized or the interface temperature had been constant for at least 100 min. Following the end of the simulation, the maximum and minimum interface heights were determined. If the interfacial height difference $\Delta h = h_{max} - h_{min}$ is significantly above 0, the simulation has destabilized. Using the interfacial height difference as the stability indicator yields Fig. 2. The critical velocity for a transition from a stable planar ice front to a cellular morphology decreases roughly linearly with the particle concentration in the suspension, as also observed in [25]. Furthermore, the higher temperature gradient shows an increase of the critical velocity. However, its effect seems to be nonlinear, as factor of 16 in the gradient only yields a factor of 2 increase in the critical velocity. This is in contrast to classical theory on the stability of planar fronts [17, 24] in which the gradient enters the stability condition



(a) Stability map for $G = 1.5\,\mathrm{K/mm}$.

(b) Stability map for $G = 24\,\mathrm{K/mm}$.

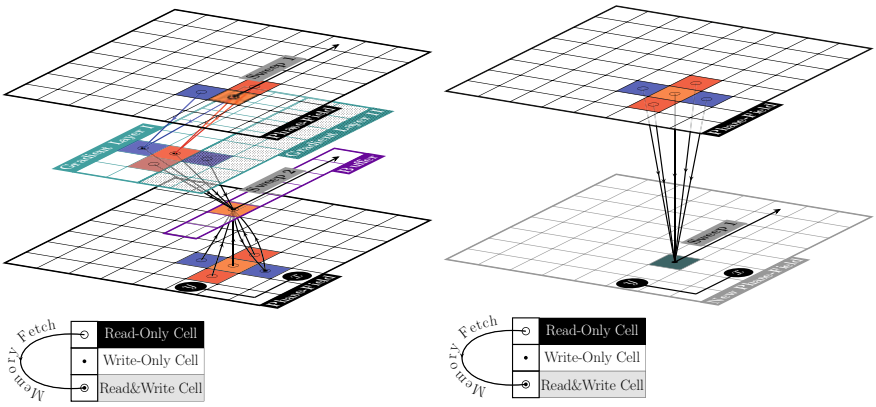
Fig. 2 Stability maps for two temperature gradients. The dots indicate stable simulations and the crosses unstable simulations. It can be seen that as in the experiments, higher suspension concentrations cause the stability limit to shift to lower velocities. Furthermore, a higher temperature gradient increases the critical velocity for the transition from a planar interface to a cellular interface

linearly. Further investigations will be conducted to determine the cause of observed nonlinearity.

4.2 Optimization Techniques for an SOFC-Anode Model

Solide oxide fuel cells (SOFCs) are one of the most promising devices which enable efficient, decentralized and low pollution energy supply. Gaining a long lifetime and corresponding cost-efficient operation is one of the most challenging tasks in the field of SOFC. The lifetime is commonly limited due to several causes, one of which is the degradation of anode material [11]. The SOFC-anode is comprised of a fine network of pores embedded in a solid electrolyte and metallic compound. State-of-the art materials for the anode are Yttria-stabilized Zirconia (YSZ) and nickel which act as the ion and electron conductor, respectively. Since SOFCs are operated at high temperatures (700–1000 °C), nickel diffuses significantly. The corresponding change in the anode microstructure is commonly held responsible for a lowering of the SOFC performance [4, 18].

Due to the long period of operation associated with SOFCs, a correspondingly long simulation time is required in order to successfully predict long-term degradation. Therefore, an optimization of the solver towards a reduced runtime was carried out. The phase-field evolution equations are solved using finite differences and integrated in time by an explicit Euler scheme. The optimized solver largely relies on the availability of vectorized floating point computing units, which is an intrinsic feature



(a) Original Solver (b) Optimized Solver

Fig. 3 Schematic of the memory access patterns in a 2D-domain for the original solver (a) and the optimized solver (b) for a sequential sweep (non-vectorized). Open and closed circles denote a read and write operation, respectively. Lines and corresponding arrows indicate the chronology of memory operations whereas colored cells belong to the stencils. Note that for the original solver there exists only a single phase-field, which is shown at the top and bottom for sake of clarity

of modern CPUs. Additionally, it exploits the specific mechanism of memory caching and fetching in the hardware by massively reusing data elements in order to keep the number of loads and stores minimal as well as conforming to alignment requirements.

For the herein discussed solvers, the memory access patterns are illustrated in Fig. 3 for a pure phase-field sweep. Note that in the grand-potential model, the phase-field sweep additionally requires information from the μ-fields which is omitted here for the sake of simplicity. The illustration is also restricted to a 2D-setup, while the underlying principles are easily adapted for a 3D-setup. As a starting point, the original solver design is illustrated in Fig. 3 (a). The original general-purpose solver is optimized towards the minimization of overall memory requirements and a reduced number of floating point operations by relying on only a single phase-field (shown black) and the use of a gradient layer (teal), respectively. Let N_x and N_y denote the dimensions of the simulation grid, then the overall memory requirements are approximately $N_x \times N_y + 5 \times N_x$, which include the gradient layers ($4 \times N_x$, teal) and the phase-field buffer (N_x, purple). The domain is updated consecutively within two subsequent sweeps in x-direction, where the first sweep calculates the current gradients and stores the x and y-component for the current y-coordinate in one of the gradient-layers. The other gradient-layer contains the gradients of the previous y-layer which are swapped after the sweep of each layer. This guarantees that gradients are calculated only once for each layer. After the gradient sweep is finalized for the current y-coordinate, the update of the phase-field is achieved by the use of a phase-buffer. Here, the previously calculated gradients are used as well as additional information from the phase-field (shown below). Note that during the

sweep, the updated value is stored solely into the buffer. The storage into the phase-field is *delayed* such that the value is written into the phase-field from the buffer at the next y-layer. This prevents the unwanted use of updated phase-field values in the current sweep. While the original solver elegantly reduces the overall memory requirement, its memory access patterns are complex. Moreover, it strictly relies on a granular chronology of the cell-updates which inherently prevents from automatic vectorization by the compiler. Note that, the original solver allows for a variety of model variants being solved which requires numerous evaluations of conditionals (e.g. *if/else* statements) and hence is clearly disadvantageous in terms of performance. Additionally, the massive use of non-inlined functions calls within the kernel hampers any further optimization techniques such as arithmetic reordering. At least two reads from the phase-field are required which results in thirteen read operations (number of open circles) and four write operations (number of closed circles) per cell if the memory remains fully cached during the calculation procedures.

In contrast to this, a *double-buffering* technique is utilized in the optimized solver. Hereby, the overall memory requirement is sacrificed for a reduced number of memory fetches. The minimum memory requirement is $2 \times N_x \times N_y$. Note that, by the use of only a single sweep the number of load operations is reduced to only five while a single store operation suffices. Therefore, the number of fetch operations could largely be reduced. While the optimized version may require more gradient calculations compared to the original solver this is not seen as a serious drawback since the operations may easily be reordered either by the compiler or manually into efficient fused-multiply-add (FMA) operations. Another major advantage of the *double-buffering* design is the ease of applying a vectorization on it. A manual vectorization of the solver is illustrated in Fig. 4. Here, the vectorization is applied in x-direction to calculate the updated values of subsequent cells in a single kernel call. Padding the simulation domain in order to guarantee a divisibility by the vector width furthermore simplifies the main loop and obviates any further conditionals in the solver. Note that computed cell-vectors (red) and y-neighbors (blue) are always properly aligned to the vector width in memory. Accordingly, the solver also benefits from proper memory alignment which likely reduces the number of cache misses during the solver run. In Fig. 4 the number of kernel calls was reduced from 80 to 24 which results in a theoretical speedup of approximately 3.3. The maximum achievable speedup is equal to the vector-width (here four) and can be reached if the domain is appropriately divisible.

Both of the solvers are parallelized by means of the message passing interface (MPI) and subdivided into several domains each being independently handled by individual CPU cores. A three-dimensional domain of a total of $300 \times 300 \times 185$ cells has been used for a case study to measure the solvers performance. In a *strong-scaling* study, multiple runs of the same domain on an increasing number of compute cores and corresponding nodes have been performed. The scaling runs were conducted on the ForHLR II at the SCC Karlsruhe. Both solvers were compiled with the Intel compiler version 18.0 and using Intel-MPI 2018. The strong-scaling results are shown in Fig. 5. Here, a sublinear scaling for both of the original and vectorized solver is observed in the left sub-figure (Fig. 5a). However, compared to the original

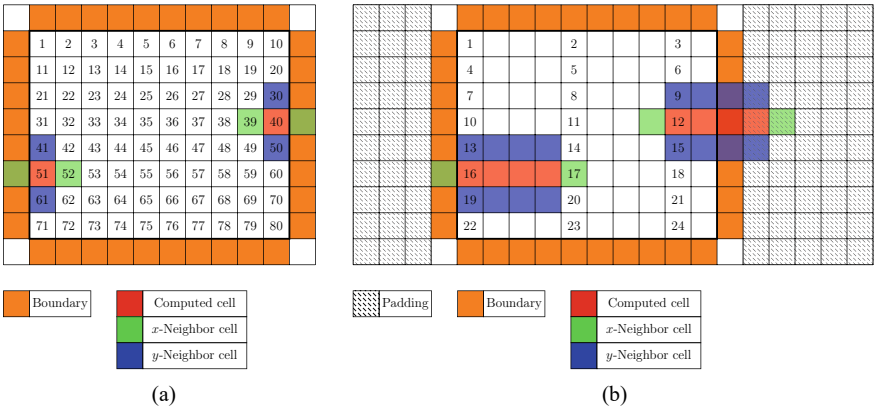


Fig. 4 Sequence of kernel calls and memory layout in the sequential solver (a) and the vectorized solver (b)

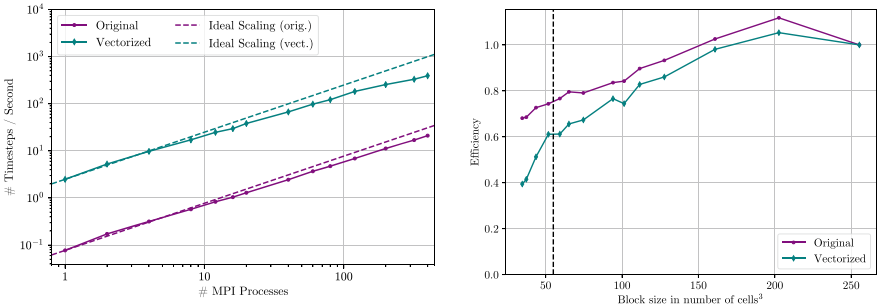


Fig. 5 Strong scaling behavior measured in terms of performance (a) and efficiency (b) as the number of compute cores (MPI processes) is increased. In (b) the efficiency is plotted over the block size per worker with the vertical dashed line indicating the production run conditions at 55^3 voxel cells

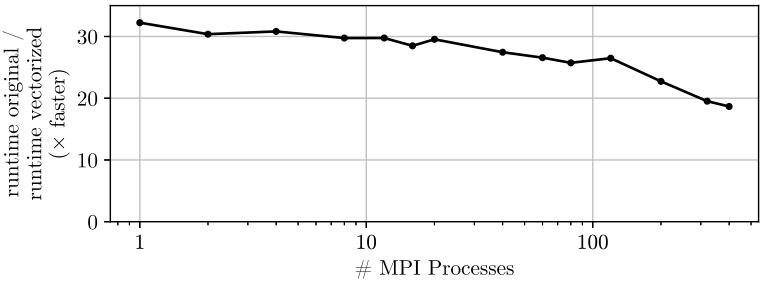
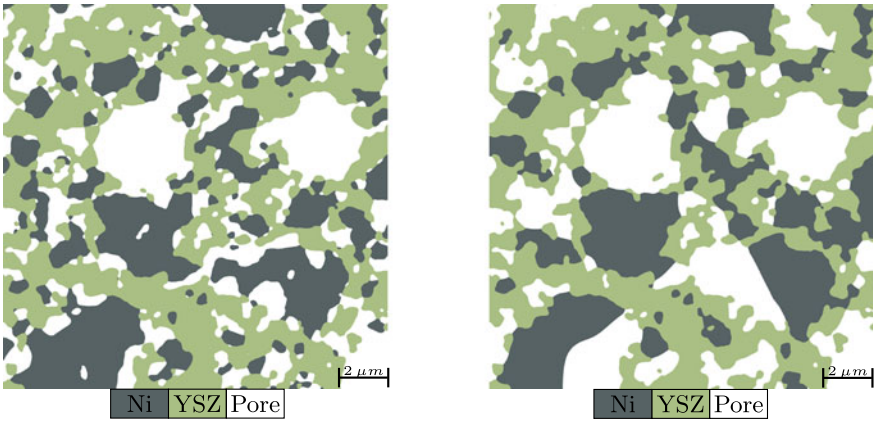


Fig. 6 Performance gain of the computation by replacing the original solver with the vectorized one in a strong-scaling study



(a) Initial microstructure (b) Final microstructure

Fig. 7 Two-dimensional cross-sections of a phase-field simulation on nickel coarsening in an SOFC-anode for the initial setup (a) and after the simulation finalized (b)

one, the vectorized solver does scale perceivably worse. This becomes even more visible if one calculates the efficiency for both of the solvers (Fig. 5b) (note the different abscissa). The efficiency for a simulation with N_{cores} number of cores is the ratio $t_{\text{ideal}}/t_{N_{\text{cores}}}$, where $t_{N_{\text{cores}}}$ and $t_{\text{ideal}} \equiv t_1/N_{\text{cores}}$ represent actual simulation time and the simulation time in case of an ideal scaling, respectively. Both of the solvers show initially an as-ideal or super-ideal behavior which belongs to a small number of compute cores (<10) on a single node. With increasing number of compute cores, both solver suffer from an efficiency decline, while the decrease in efficiency is more strongly pronounced for the optimized solver. In the case of maximum resources (400 CPU cores), the efficiency of the original and vectorized solvers are about 68% and 39%, respectively. From that point of view one might infer that the original solver is advantageous. However, taking into account the absolute compute times, which can also be deduced from Fig. 5a, it is seen that the optimized solver is at least one order of magnitude faster than the original one. The corresponding performance gain that was achieved in the study by the application of the vectorized solver is shown in Fig. 6. Here, it can be seen that at best the optimized solver is 32 times faster which was reached on a single node. With increasing number of nodes, the gain in performance considerably decays. However, in the case of maximum parallelization (400 CPU cores), the vectorized solver is still more than 18 times faster than the original one. Since the computation time in the vectorized solver is much smaller, the communication takes a comparatively long period of time. With increased parallelization, the communication time increases which may explain the reduced scalability of the vectorized solver. A typical choice of simulation block-size is additionally shown in Fig. 5b which belongs to an efficiency larger than 60% for the optimized solver.

In Fig. 7 two-dimensional cross-sections of the microstructure corresponding to the initial state and at the end of the simulation run are shown. First of all, as desired, the YSZ-structure remains invariant during the simulation run. In contrast to this, the coarsening of the nickel particles is strongly pronounced. In the initial setup, the nickel particles are irregularly shaped and of relatively small size. In the final state, rounded and/or flat surfaces dominate the morphology of the nickel particles while the average particle size has apparently increased. Interestingly, due to the invariance of the YSZ-structure, further coarsening of few nickel particles could be significantly suppressed. To obtain deeper understanding of the underlying phenomena, in addition to the qualitative results presented here, a future publication will be devoted to a detailed and quantitative investigation of nickel coarsening in SOFC-anodes. This study will largely benefit from the herein discussed solver optimizations.

4.3 Time-to-Solution and Scalability of Different Time Integration Schemes

Parts of the results presented in this section are based on work published by some of the authors [6]. Not included in that reference but included here are the small scaling results which led to the choice of time integration schemes scaled to the full machine in [6].

Since the timescales of Eq. (1) (ϕ) and Eq. (3) (μ) may differ substantially, employing different time integration schemes for the equations may be useful. Specifically, the timescale of Eq. (3) is typically below that of Eq. (1). Equation (3) is a linear diffusion-reaction equation which allows the first-order backward Euler scheme to be efficiently applied. With Eq. (1) the non-convex contribution from the bulk potential w would enforce a time step restriction, removing the advantage of implicit schemes. Hence Eq. (1) is integrated explicitly with the first-order forward Euler scheme.

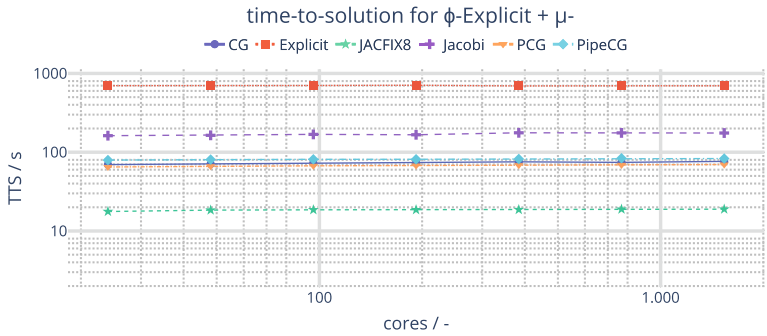
The system of equations resulting from discretizing Eq. (3) with the backward Euler scheme is solved with different methods in order to determine the most appropriate one: Jacobi, conjugated gradients (CG), Jacobi-preconditioned conjugated gradients (PCG), and pipelined Jacobi-preconditioned conjugated gradients [5] (PipeCG). All of these schemes require global communication for the residual which is a potential bottleneck for scalability. In the case of the Jacobi scheme, the residual is only a stopping condition and not required for the next iteration. Hence the number of iterations required to reach the prescribed residual norm is analyzed. After an initial period of higher iterations counts, the necessary number of iterations was generally below 8 per time step for the rest of the simulation. Therefore a Jacobi scheme with a fixed number Z of iterations (JACFIXZ) is additionally implemented. Since this scheme is by nature not certain to converge all the time a runtime kernel switching is implemented as well. First, the full Jacobi scheme is run until the number of iterations has become sufficiently low. Afterwards, the kernel is switched to JACFIXZ and low-frequency checks on the residual are conducted. If such a check fails, the full

Jacobi scheme is employed again until another region of low iterations is reached. During the scaling runs no such kernel switching was necessary.

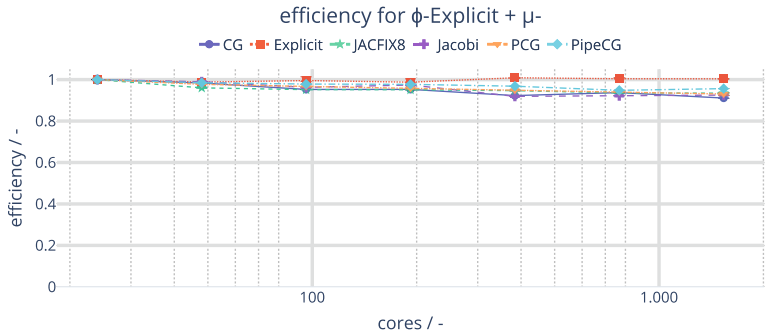
For the performance measurements the domains are constructed such that all MPI processes have equal load with a representative order parameter distribution as in production results. Each MPI process calculates the updates for 80^3 voxel cells, i.e. a weak scaling experiment is conducted. This always includes the explicitly integrated evolution for ϕ and the different time integration schemes for μ. In order to measure time-to-solution (TTS) a fixed simulation time is established as the goal. Since the implicit schemes show much better TTS behavior, the explicit scheme is not calculated to this goal time, but rather is linearly extrapolated from roughly 1/6 of the simulation time. This is done for simulations above 1 536 cores in order to reduce the computational cost of scaling the purely explicit scheme. Within the scaling from 24 to 1 536 cores the extrapolation is validated and found to have an error less than 0.1 %. The implicit schemes always calculate 100 time steps, the extrapolated explicit scheme 1000 and the non-extrapolated explicit scheme 6400 time steps. The time step width of the explicit scheme corresponds to the von Neumann stability limit of Eq. (3). The time step width of the implicit scheme is set to 64 times that of explicit scheme. The reported TTS includes timing for the whole application with I/O, but without (de-)initialization.

Figure 8a shows the TTS up to 1 536 cores for all considered schemes. The TTS is plotted on a logarithmic scale to show the massive advantage of implicit schemes in reaching the prescribed simulation time; whereas the explicit scheme has a runtime close to 700 s, all of the implicit schemes clock in at a runtime below 200 s. The JACFIX8 scheme reaches a runtime of around 20 s, which implies a speedup of 35 relative to the purely explicit schemes. As already suggested by the nearly horizontal lines in Fig. 8a, the efficiencies relative to a single node (24 cores) of the schemes depicted in Fig. 8b are high and stay above 90 % up to the maximum considered core count.

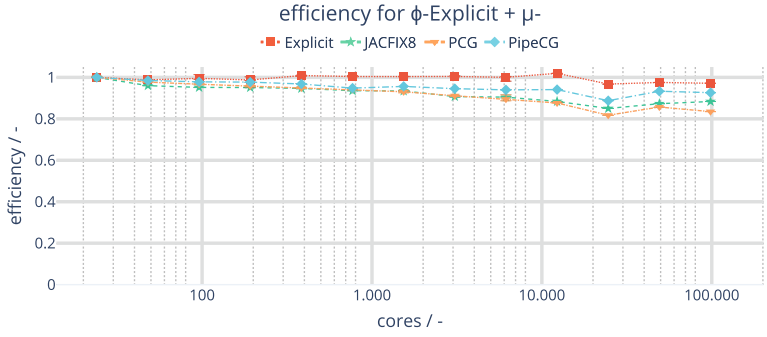
Based on these results the Jacobi scheme can easily be excluded from further investigation, as it shows both a worse TTS and slightly worse scaling behavior than the other schemes. For the schemes in the CG group, the standard conjugated gradient scheme is excluded as it is likely to scale worse than its brethren PipeCG and PCG. This leads to Fig. 8c in which the remaining schemes (purely explicit with extrapolation, JACFIX8, PipeCG and PCG) were scaled up to 98304 cores on the Hazel Hen. A significant further drop in efficiency is observed for all implicit schemes, with the PCG showing at worst an efficiency of about 80 % at 98304 cores. As expected, the PipeCG scheme shows better scaling behavior than the PCG scheme since it allows global communication to be done asynchronously. Somewhat surprisingly the JACFIX8 scales worse than the PipeCG scheme even though it does not require global communication. This can be explained by the very low computational effort of the JACFIX8 scheme per iteration within a time step, after which a ghost layer exchange is necessary. Hence the drop in efficiency of the CG schemes can be attributed to global communication being the bottleneck, whereas for the JACFIX8 scheme the local communication seems to be the limiting factor.



(a) time-to-solution up to 1536 cores



(b) efficiency up to 1536 cores



(c) efficiency up to 98304 cores

Fig. 8 Whole-application efficiency and time to solution plots of multi-node weak scaling of 80^3 voxel cells per core. (a and b) up to 1536 cores corresponding to 64 nodes and (c) up to 98304 cores corresponding to 4096 nodes on the Hazel Hen

5 Conclusion

Applications of a general multiphase-field model based on the grand potential approach in the context of high performance computing have been presented. Within the document, three different topics are discussed which are (1) the stability during freeze-casting of ice (2) nickel coarsening in an Ni-YSZ solide-oxide-fuel-cell (SOFC) anode and (3) sintering of alumina. During freeze-casting the effect of concentration in the liquid and the temperature gradient on the stability of a planar growth front was investigated. The critical velocity beyond which the planar front destabilized was observed to depend roughly linearly on the concentration in the liquid, which matches experimental results and theory. However, the temperature gradient is observed to have a nonlinear influence on the critical velocity, which is in contrast to theory. The cause of this nonlinearity will be investigated in future work. In the case of nickel coarsening in SOFC-anodes, details on the implementation and optimization strategies towards a reduced runtime of the MPI-parallelized solver were presented. A comparative strong-scaling study was carried out to investigate the scaling and performance gain of the optimized solver for an increased number of compute cores. It could be demonstrated, that the runtime of the optimized solver is more than one order of magnitude faster than the original general-purpose solver. The performance gain was achieved through a design towards reduced number of memory fetches and by a manual SIMD-vectorization of the code. First qualitative simulation results show a pronounced coarsening of nickel in the SOFC-anode. For the sintering of alumina, the influence of different numerical solution methods for an implicit backward-Euler scheme on the scalability and performance of the solver at hand are demonstrated. It is seen that all methods scale well up to almost 100 thousand compute cores. Differences arising from the communication requirements of the methods are discussed which may explain the observed differences in performance.

Acknowledgements Part of this work was performed on the supercomputer ForHLR funded by the Ministry of Science, Research and the Arts Baden-Württemberg and by the Federal Ministry of Education and Research. Part of this work was performed on the national supercomputer Cray XC40 Hazel Hen at the High Performance Computing Center Stuttgart (HLRS) under the grant number pace3d. The authors gratefully acknowledge financial support of research on SOFC anode development by the German Federal Ministry for Economic Affairs and Energy (BMWi) within the project KerSOLife100 under Funding No. 03ET6101A. The authors gratefully acknowledge financial support of the parallel code development of the PACE3D package and of ice formation during freeze casting by the Deutsche Forschungsgemeinschaft (DFG) under the grant numbers NE 822/31-1 (Gottfried-Wilhelm Leibniz prize) and NE 822/9-2, respectively. The authors thank for the financial support of general modelling concept for chemo-mechanically driven microstructure evolutions by the Helmholtz association through the programme "Energy Efficiency, Materials and Resources" (EMR). The authors thank for the financial support within the projects SKAMPY (Ultra-scalable multiphysics simulations for solidification processes in metals) and PoroSan (Development of a porosity model for anisotropic sand structures for the description of adsorptive processes during ground water purification) funded by Bundesministerium für Bildung und Forschung (BMBF).

References

1. A. Choudhury, B. Nestler, Grand-potential formulation for multicomponent phase transformations combined with thin-interface asymptotics of the double-obstacle potential. Phys. Rev. E **85**(2), 021602 (2012)
2. J.A. Dantzig, M. Rappaz, *Solidification*. (EPFL press, 2009)
3. B. Echebarria, R. Folch, A. Karma, M. Plapp, Quantitative phase-field model of alloy solidification. Phys. Rev. E **70**(6), 061604 (2004)
4. A. Faes, A. Hessler-Wyser, D. Presvytes, C.G. Vayenas, J. Van herle. Nickel-zirconia anode degradation and triple phase boundary quantification from microstructural analysis. Fuel Cells **9**(6), 841–851 (2009)
5. P. Ghysels, W. Vanroose, Hiding global synchronization latency in the preconditioned conjugate gradient algorithm. Parallel Comput. **40**(7), 224–238 (2014). 7th Workshop on Parallel Matrix Algorithms and Applications
6. H. Hierl, J. Hötzer, M. Seiz, A. Reiter, B. Nestler, Extreme scale phase-field simulation of sintering processes, in *2019 IEEE/ACM 10th Workshop on Latest Advances in Scalable Algorithms for Large-Scale Systems (ScalA)*, pp. 25–32 (2019)
7. J. Hötzer, A. Reiter, H. Hierl, P. Steinmetz, M. Selzer, B. Nestler, The parallel multi-physics phase-field framework pace3d. J. Comput. Sci. **26**, 1–12 (2018)
8. J. Hötzer, M. Seiz, M. Kellner, W. Rheinheimer, B. Nestler, Phase-field simulation of solid state sintering. Acta Mater. **164**, 184–195 (2019)
9. J. Hötzer, M. Jainta, P. Steinmetz, B. Nestler, A. Dennstedt, A. Genau, M. Bauer, H. Köstler, U. Rüde, Large scale phase-field simulations of directional ternary eutectic solidification. Acta Mater. **93**, 194–204 (2015)
10. J. Hötzer, O. Tschukin, M. Ben Said, M. Berghoff, M. Jainta, G. Barthelemy, N. Smorchkov, D. Schneider, M. Selzer, B. Nestler, Calibration of a multi-phase field model with quantitative angle measurement. J. Mater. Sci. **51**(4), 1788–1797 (2015)
11. S.P. Jiang, S.H. Chan, San Ping Jiang and Siew Hwa Chan, A review of anode materials development in solid oxide fuel cells. J. Mater. Sci. **39**(14), 4405–4439 (2004)
12. A. Karma, Phase-field formulation for quantitative modeling of alloy solidification. Phys. Rev. Lett. **87**(11), 115701 (2001)
13. A. Karma, W.-J. Rappel, Quantitative phase-field modeling of dendritic growth in two and three dimensions. Phys. Rev. E **57**(4), 4323–4349 (1998)
14. S.G. Kim, D.I. Kim, W.T. Kim, Y.B. Park, Computer simulations of two-dimensional and three-dimensional ideal grain growth. Phys. Rev. E **74**(6), 061605 (2006)
15. Y. Liu, W. Zhu, K. Guan, C. Peng, W. Jianqing, Freeze-casting of alumina ultra-filtration membranes with good performance for anionic dye separation. Ceram. Int. **44**(10), 11901–11904 (2018)
16. N. Moelans, A quantitative and thermodynamically consistent phase-field interpolation function for multi-phase systems. Acta Mater. **59**(3), 1077–1086 (2011)
17. W.W. Mullins, R.F. Sekerka, Stability of a planar interface during solidification of a dilute binary alloy. J. Appl. Phys. **35**(2), 444–451 (1964)
18. M.H. Pihlatie, A. Kaiser, M. Mogensen, M. Chen, Electrical conductivity of Ni-YSZ composites: degradation due to Ni particle growth. Solid State Ionics **189**(1), 82–90 (2011)
19. M. Plapp, Unified derivation of phase-field models for alloy solidification from a grand-potential functional. Phys. Rev. E **84**, 031601 (2011)
20. M. Plapp, Unified derivation of phase-field models for alloy solidification from a grand-potential functional. Phys. Rev. E Stat. Nonlinear Soft Matter Phys. **84**(3), 1–36 (2011)
21. H. Schoof, J. Apel, I. Heschel, G. Rau, Control of pore structure and size in freeze-dried collagen sponges. J. Biomed. Mater. Res. **58**(4), 352–357 (2001)
22. S.W. Sofie, F. Dogan, Freeze Casting of Aqueous Alumina Slurries with Glycerol. **64**(188747), 1459–1464 (2001)
23. I. Steinbach, F. Pezzolla, A generalized field method for multiphase transformations using interface fields. Phys. D Nonlinear Phenomena **134**(4), 385–393 (1999)

24. W.A. Tiller, K.A. Jackson, J.W. Rutter, B. Chalmers, The redistribution of solute atoms during the solidification of metals. Acta Metall. **1**(4), 428–437 (1953)
25. T. Waschkies, R. Oberacker, M.J. Hoffmann, Investigation of structure formation during freeze-casting from very slow to very fast solidification velocities. Acta Mater. **59**(13), 5135–5145 (2011)
26. U.G.K. Wegst, M. Schecter, A.E. Donius, P.M. Hunger, Biomaterials by freeze casting. Philos. Trans. R. Soc. A Math. Phys. Eng. Sci. **368**(1917), 2099–2121 (2010)

High-Performance Shape Optimization for Linear Elastic Models of Epidermal Cell Structures

Jose Pinzon, Martin Siebenborn, and Andreas Vogel

Abstract We employ parallel shape optimization to find optimal configurations of human skin cell compounds minimizing the stored energy. To this end, we model linear elastic mechanical properties of epidermal cell structures including additional constraints for space-filling designs with minimal surfaces. The large distributed-memory cluster *Hazel Hen* is used to simulate these three-dimensional domains at a cellular level. Several benchmark tests are carried out using a gradient-penalized shape optimization algorithm comparing the influence of optimization weights. In addition, an enhanced regularization for locally nonsingular deformation mappings is presented for the two-dimensional setting which significantly reduces the number of required optimization steps while retaining very thin inter-cellular channels. Results for weak scaling studies are shown for up to 3 billion degrees of freedom and 12,288 cores with close to ideal speedup.

1 Introduction

The main goal of our project is to understand the structural layout of human skin at a cellular level with respect to its mechanical properties. To this end, we employ shape optimization to study the influence of optimality objectives guided by the principle

J. Pinzon (✉) · A. Vogel
High Performance Computing in the Engineering Sciences, Ruhr University Bochum, Universitätsstraße 150, 44801 Bochum, Germany
e-mail: jose.pinzonescobar@rub.de

A. Vogel
e-mail: a.vogel@rub.de

M. Siebenborn
Department of Mathematics, University Hamburg, Bundesstraße 55, 20146 Hamburg, Germany
e-mail: martin.siebenborn@uni-hamburg.de

W. E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '20*,
https://doi.org/10.1007/978-3-030-80602-6_38

of minimal energy and compare the resulting cell arrangements with the settings observed in nature.

Overviews for skin models can be found in [8, 9, 11, 14] and most models employ a continuum formulation with constitutive equations. However, the geometrical arrangement of cells has a strong effect on the skin properties. The influence on its permeability has been investigated, e.g., in [12, 28]. In such studies, the arrangement in the *stratum corneum*, the uppermost layer of the epidermis, is captured by an apriori assumed geometrical representation such as brick-and-mortar [5] and tetrakaidecahedral (TKD) [3] models. Interestingly, TKD-shaped cell arrangements possess an optimal ratio between cell volume and outer surface also known as Kelvin's conjecture [24]. This suggests that the cell arrangement is guided by optimality principles. We therefore used these constraints and the minimization of elastic energy to investigate the impact on the optimal cell arrangement in 2d [22] and 3d [13] domains. Notably, no prior information other than cell levels with different mechanical properties embedded in an outer surrounding material are required to resemble the characteristic shape configurations of the stratum corneum. For the biological domains, contacts and overlaps of cells are prevented and a very thin lipid channel separates the cells. These conditions are introduced into our modeling and optimization scheme by appropriate regularization.

In order to find the optimal arrangements, methods from the field of PDE-constrained shape optimization are used. An overview about the topic can be found, e.g., in [4]. The objective in our simulation is to reshape and relocate internal interfaces in order to minimize the elastic energy under an applied tensile force. Detailed information about the mathematical methodology including the selection of appropriate inner product and shape spaces can be found in [19, 20, 23] and the application to cellular composites in [22]. In order to regularize the mesh movement, a penalization of high-valued deformation gradients is employed and helps to maintain the mesh quality to the required extend. For efficiency reasons, we use the geometric multigrid method [6] to solve the linear systems of equations. For the parallel scalability, the hierarchically distributed multigrid approach [16, 27] is employed, which has proven efficient for human skin simulations in prior works [15, 18, 25, 26].

2 Mathematical Methodology

We solve the PDE-constrained shape optimization problem [13, 22] given by

$$\begin{aligned} \min_{\Gamma_{\text{int}}} \quad & J(u,\Omega) := \frac{\nu_{\text{elast}}}{2}\int_{\Omega}\sigma(u):\nabla u\,dx + \nu_{\text{vol}}\int_{\Omega_{\text{out}}} 1\,dx + \nu_{\text{peri}}\int_{\Gamma_{\text{int}}} 1\,ds \\ \text{s.t.} \quad & \int_{\Omega}\sigma(u):\nabla w\,dx = \int_{\Gamma_{\text{top}}} f\cdot w\,ds \quad \forall\, w. \end{aligned} \tag{1}$$

Algorithm 1 Gradient-penalized optimization algorithm from [22]

1: Choose initial domain Ω^0; choose ν_{elast}, ν_{vol}, ν_{peri}; choose ν_{penalty}, b, t
2: **for** $k = 0, 1, 2, \ldots$ **do**
3: Compute displacement field u^k, i.e. solve the elasticity problem, for the current Ω^k
4: Compute shape derivative $dJ(\Omega)[w]$, i.e. assemble $dJ(\Omega^k) = \nu_{\text{elast}} dJ_{\text{elast}}(u^k, \Omega^k) + \nu_{\text{vol}} dJ_{\text{vol}}(\Omega^k) + \nu_{\text{peri}} dJ_{\text{peri}}(\Omega^k)$
5: Reset shape derivative in interior, i.e. set $dJ(\Omega^k)[\phi] = 0$ if $\text{supp}(\phi) \cap \Gamma_{\text{int}} = \emptyset$
6: Compute descent direction v, i.e. solve $g(v, w) = dJ(\Omega^k)[w]$ for all $w \in H_0^1(\Omega, \mathbb{R}^d)$
7: Update shape: $\Omega^{k+1} = \{x + tv(x) : x \in \Omega^k\}$
8: **end for**

Here, u denotes the displacement of a linear elastic material Ω, if a force f is applied to the upper boundary Γ_{top}. The objective function is composed of the elastic energy multiplied by a weighting factor ν_{elast}, and the geometrical terms that represent the minimization of the outer volume Ω_{out} and of the cell surfaces Γ_{int}, weighted by the terms ν_{vol} and ν_{peri}, respectively. The optimization variable is the shape and location of internal interfaces Γ_{int}, which separate materials with different properties, i.e., we treat the skin as a composite material with different properties for skin cell layers and fill-in material.

In the following, $dJ(\Omega)[w]$ denotes the shape derivative evaluated in direction w, cf. [22, 23]. The approach in [21] is pursued to represent the shape derivative w.r.t. appropriate inner products. A common choice is to utilize the bilinear form of linear elasticity in order to determine the gradient v by solving

$$a(v, w) := \int_\Omega \sigma(v) : \nabla w \, dx = dJ(\Omega)[w], \quad \forall w \in H_0^1(\Omega, \mathbb{R}^d). \tag{2}$$

In addition, the biological aspects of epidermal cell interaction have to be incorporated into the modeling. This includes to avoid contact between the cells by maintaining a thin inter-cellular channel between them, as well as to prevent disappearing subdomains due to the shape movement. This is achieved by a modification of the inner product to

$$a(v, w) + \nu_{\text{penalty}} \int_\Omega \left(\nabla v : \nabla v - b^2\right)_+ \nabla v : \nabla w \, dx = dJ(\Omega)[w] \quad \forall w. \tag{3}$$

The additional penalty term promotes descent directions whose norm of the deformation gradient does not exceed a threshold b. If effective, the bilinear form switches to a nonlinear behavior. The final algorithm to compute the optimal shapes is shown in Algorithm 1. The weighting factors and threshold are found empirically and its influence is the subject of this report.

3 Numerical Results

All simulations are carried out using UG4 [27] which is a simulation framework for PDE-based models on high-performance computers. The parallel scalability of UG4 is investigated and documented in [16, 17, 25]. The Finite Element method (FEM) has been used to discretize the involved PDEs of the shape optimization problems. The computational meshes are created using ProMesh [1]. ParaView [2] is utilized for visualization of the cell arrangements in the deformed configuration. The simulations are carried out using one MPI process per core. The 3d model is refined two times, starting at 2,953 tetrahedral elements up to 188,992 in the highest level of refinement. It features five cell levels with 35 cells in total. Due to simplified boundary conditions, only inner cells deform to TKD shapes. See [13] for details and a visualization of computed inner TKDs.

3.1 Influence Study for Weighting Factors and Threshold

Employing several case studies, we illustrate the effects of the weighting factors ν_{elast}, ν_{peri}, ν_{vol} in the objective function given in Eq. (1) as well as the threshold factor b and ν_{penalty} in Eq. (3) on the final epidermal cell arrangement. The influences are studied with respect to parameters used for the elastic energy minimization in [13] with an approximately space-filling TKD cell design. The modifications of the constants have effects on the total number of steps, geometric configuration, and therefore on the final inner cell volume. Optimization steps are performed as long as a reasonable mesh quality is maintained and the simulation is aborted once the quality leads to an insufficient multigrid solver convergence.

3.1.1 Influence of the Threshold for the Deformation Gradient

The threshold value b limits the Frobenius norm of the deformation gradient. This controls the node displacement within each optimization step. As a consequence, deformations that could lead to contact between inclusions are prevented maintaining inter-cellular channels and retaining mesh quality. In order to test the influence of this parameter, we consider two configurations with different threshold values b given by

$$\begin{aligned} &\nu_{\text{elast}} = 10^5 && \nu_{\text{vol}} = 10^4 && \nu_{\text{peri}} = 10^3 \\ &\nu_{\text{penalty}} = 5 \cdot 10^7 && b \in \{10^{-2}, 10^{-5}\} && \text{refinements} = 2. \end{aligned}$$

For the specific weighting factors used here, a maximum threshold turns out to be 10^{-2}. Figure 1 shows the cell arrangement for the two configurations. The results are geometrically almost equivalent. The difference between the two optimization

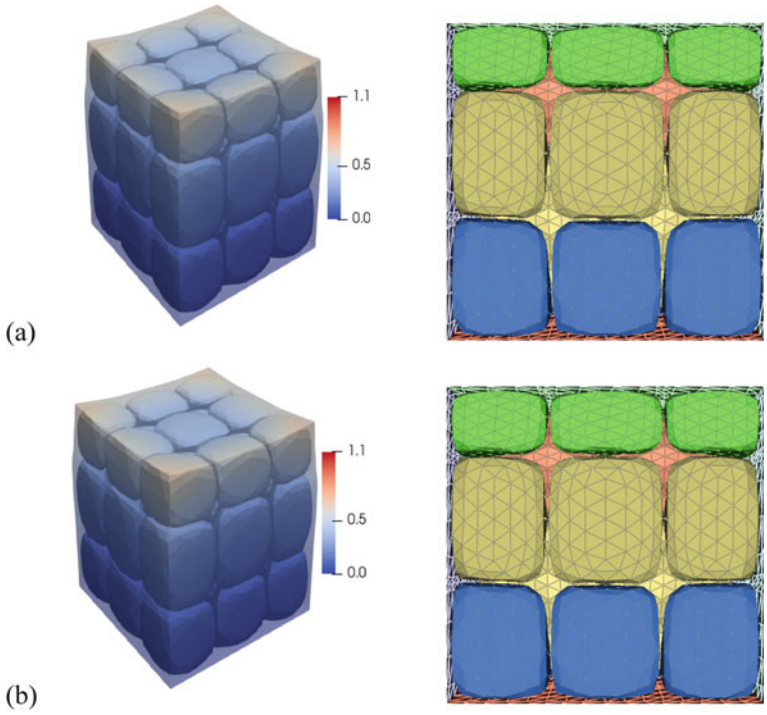


Fig. 1 Influence of deformation gradient threshold value: Deformed (left) and reference (right) configurations in optimization step 37 for: (a) $b = 10^{-2}$, (b) $b = 10^{-5}$

runs is visible in different numbers of iterations. While $b = 10^{-2}$ results in 37 optimization steps, the optimization with $b = 10^{-5}$ requires 44 gradient descent steps.

3.1.2 Influence of Penalization Weight ν_{penalty}

The factor ν_{penalty} penalizes large gradients ∇v of the descent direction calculated in step 6 of Algorithm 1, if the Frobenius norm of ∇v exceeds b, cf. Eq. (3). The interaction between the two values ν_{penalty} and b can be summarized as follows: by b, a maximum of the Frobenius-norm of the deformation gradient is determined and ν_{penalty} controls how much the excess of b is penalized. From a mathematical point of view, b controls when the inner product of the shape space turns to a nonlinear behavior.

While it is shown in Sect. 3.1.1 that b only has a small effect on the number of optimization steps for a substantial reduction of three orders of magnitude, the effects of modifying ν_{penalty} are more significant. For this test we use the values

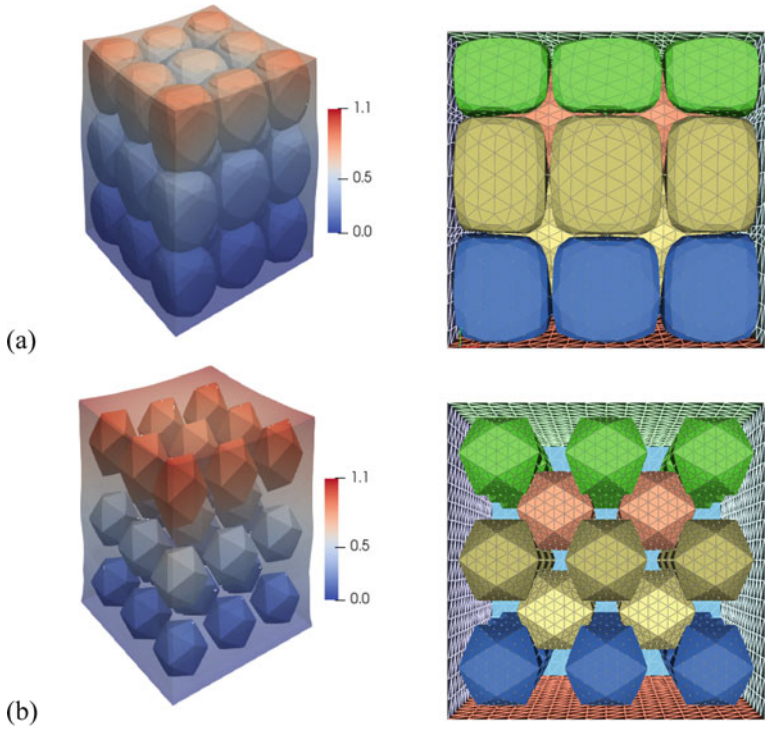


Fig. 2 Influence of ν_{penalty}: Deformed (left) and reference (right) configurations in step 2 for simulations with: (a) $\nu_{\text{penalty}} = 5 \cdot 10^4$, (b) $\nu_{\text{penalty}} = 5 \cdot 10^7$

$$\nu_{\text{elast}} = 10^5 \qquad \nu_{\text{vol}} = 10^4 \qquad \nu_{\text{peri}} = 10^3$$
$$\nu_{\text{penalty}} \in \{5 \cdot 10^4, 5 \cdot 10^7\} \qquad b = 10^{-3} \qquad \text{refinements} = 2.$$

The results are shown in Fig. 2. A reduction by three orders of magnitude in the penalty factor caused a significant decrease in the number of optimization steps. For $\nu_{\text{penalty}} = 1 \cdot 10^5$, only 2 optimization steps are possible, because the inner cell volume increases too rapidly leading to intersecting finite elements. Moreover, there are also geometrical differences in the final cell arrangement compared to the one shown in Fig. 1, where the uppermost cell level, the softer cells, are compressed by the stiffer levels.

In Fig. 2, the simulation stops before the compression is as shown in Fig. 1. The configuration with $\nu_{\text{penalty}} = 5 \cdot 10^7$ in Fig. 2(b) is still in the process of increasing the inner cell volume. This shows that, although there is a large tolerance for the range of b values, the final cell arrangement is much more sensitive w.r.t. modifications in ν_{penalty}. Moreover, the observed number of optimization steps is more sensitive in terms of ν_{penalty} than b.

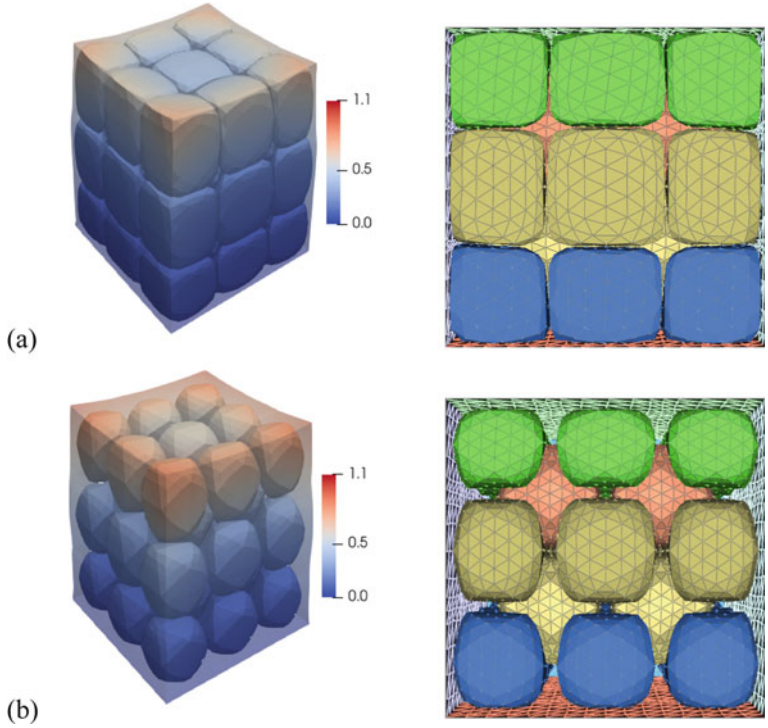


Fig. 3 Influence of ν_{vol}: Deformed (left) and reference (right) configurations in step 16 for simulations with: (a) $\nu_{\text{vol}} = 2 \cdot 10^5$, (b) $\nu_{\text{vol}} = 10^4$

3.1.3 Influence of Volume Weight ν_{vol}

The next experiment illustrates the influence of the factor ν_{vol}. In the objective function in Eq. (1), it weights the volume of the fill-in material between the cells. Minimizing this volume leads to a compression of the channels and an expansion of the cells themselves. For larger values of ν_{vol}, we expect the compression to dominate the other terms in the objective function and thus lead to a space-filling design. We test the settings

$$\nu_{\text{elast}} = 10^5 \qquad \nu_{\text{vol}} \in \{5 \cdot 10^5, 10^4\} \qquad \nu_{\text{peri}} = 10^3$$
$$\nu_{\text{penalty}} = 5 \cdot 10^7 \qquad b = 10^{-3} \qquad \text{refinements} = 2.$$

For larger volume weight, the expected results are a smaller number of gradient steps and a cell arrangement with a denser outer volume filling, but with a minor compression of the softer cells in the upper part of the domain. Figure 3 shows a comparison between the experimental settings, also cf. [13]. The larger value of ν_{vol} results in 16 steps being solved as opposed to 44 gradient steps for a smaller weight.

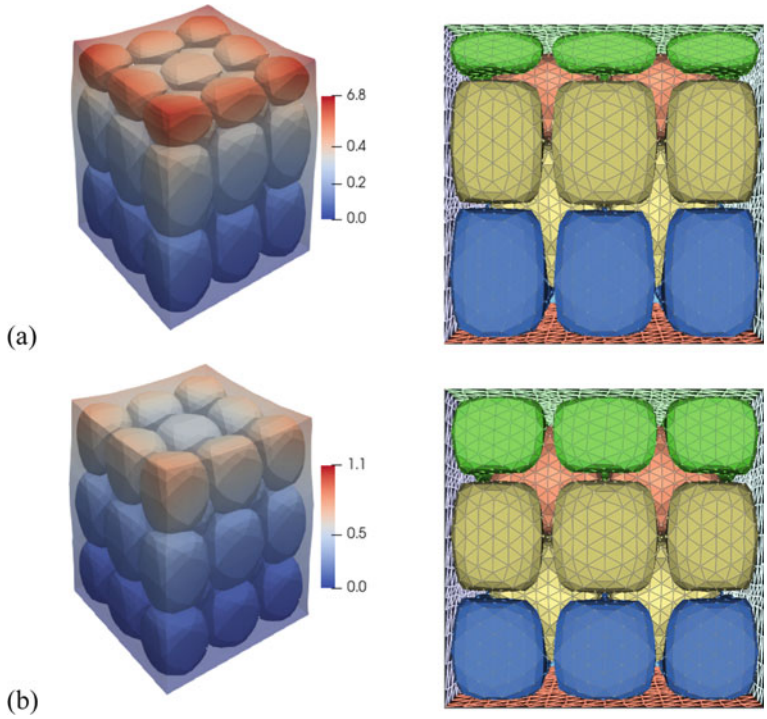


Fig. 4 Influence of ν_{elast}: Deformed (left) and reference (right) configurations in step 22 for simulations with: (a) $\nu_{\text{elast}} = 5 \cdot 10^5$, (b) $\nu_{\text{elast}} = 10^5$

For $\nu_{\text{vol}} = 10^4$, the cells remain being surrounded by a relatively thick layer of the fill-in material. Whereas, for $\nu_{\text{vol}} = 2 \cdot 10^5$, the optimization terminates with a thin layer, but without significant compression of the uppermost cell level.

3.1.4 Influence of Elastic Weight ν_{elast}

The experiments in this section illustrate the influence of ν_{elast}. This factor weights the elastic energy in the objective function in Eq. (1). In contrast to the pure geometric terms in the objective, this value is coupled with the displacement u determined by the PDE constraint. While dominating values of ν_{vol} and ν_{peri} lead to a space-filling TKD design, larger values of ν_{elast} move the focus of the final configuration towards stiffness. In the following, we present results with the configurations

$$\nu_{\text{elast}} = \{5 \cdot 10^5, 10^5\} \qquad \nu_{\text{vol}} = 10^4 \qquad \nu_{\text{peri}} = 10^3$$
$$\nu_{\text{penalty}} = 5 \cdot 10^7 \qquad b = 10^{-3} \qquad \text{refinements} = 2.$$

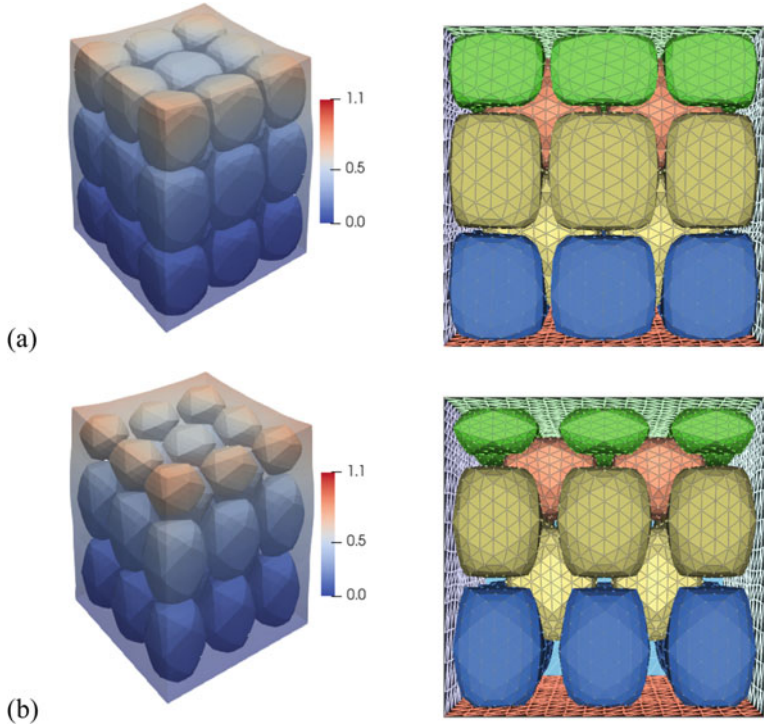


Fig. 5 Influence of geometrical weights: Deformed (left) and reference (right) configurations in step 27 for: (a) $\nu_{\text{vol}} > \nu_{\text{peri}}$, (b) $\nu_{\text{vol}} < \nu_{\text{peri}}$

In Fig. 4, the results after 22 gradient steps are shown. The larger value of ν_{elast} is reflected in a higher compression of the softer cells by the expanding stiffer ones. In this case, the optimization is dominated by the elastic properties of the cell arrangement rather than volume and perimeter minimization. Theses results also show that with a larger value of ν_{elast} the final configuration exhibits more stiffness in the tensile test, which is described by the PDE constraint.

3.1.5 Interplay Between the Geometrical Factors ν_{peri} and ν_{vol}

This section is devoted to a study of the two geometric weighting factors ν_{vol} and ν_{peri} while ν_{elast} is fixed. The associated two terms in the objective function in Eq. (1) have opposing geometrical effects. Minimizing the volume of the fill-in material leads to an increase of the cell volumes which is accompanied by an increase of their surface area. Thus, the ratio between ν_{vol} and ν_{peri} determines whether the optimized configuration shows inflating shapes or decreasing spherical cells.

As test settings, two extreme cases of the proportion between ν_{vol} and ν_{peri} are presented. The first configuration is

$$\begin{aligned} \nu_{\text{elast}} &= 10^5 & \nu_{\text{vol}} &= 10^4 & \nu_{\text{peri}} &= 10 \\ \nu_{\text{penalty}} &= 5 \cdot 10^7 & b &= 10^{-3} & \text{refinements} &= 2, \end{aligned}$$

where the volume term is four orders of magnitude larger than the surface term. The second setting presents the opposite case,

$$\begin{aligned} \nu_{\text{elast}} &= 10^5 & \nu_{\text{vol}} &= 10^3 & \nu_{\text{peri}} &= 10^4 \\ \nu_{\text{penalty}} &= 5 \cdot 10^7 & b &= 10^{-3} & \text{refinements} &= 2, \end{aligned}$$

where the surface weight is larger than the volume weighting factor.

The results are shown in Fig. 5. The larger value of ν_{vol} promotes the inner cell volume expansion, which can be seen by the larger cells after 27 gradient steps. Inverting the ratio, i.e., smaller ν_{vol} and larger ν_{peri}, results in smaller cells. This reflects the dominant influence of the surface term in the objective, that prevents the cells from expanding. In our experiments, it turns out that $\nu_{\text{vol}} > \nu_{\text{peri}}$ results in approximately space-filling cell arrangement.

3.2 Locally Injective Descent Directions

In the previous sections, we computed the results employing a weakly enforced constraint for the upper limit of the Frobenius norm of the mesh deformation. The aim of the additional term is to prevent mesh degradation by allowing only sufficiently small displacements. This strategy has also been employed in [22] and has proven to be efficiently scalable on large distributed memory clusters [13]. However, a significant number of optimization steps is required to reach the optimized shapes. In contrast, we now present results for a different approach which has been proposed for the general case in [7] and is applied here to the human skin setting. The approach enforces the condition

$$\det(\nabla \mathcal{F}) \geq b \quad \text{a.e. in } \Omega \tag{4}$$

where $\mathcal{F} = \text{id} + v$ denotes the deformation mapping. This leads to locally injective deformations and is realized by the following penalty term in the objective function

$$J_{\det}(v) := \frac{\nu_{\text{penalty}}}{2} \int_\Omega ((b - \det(\nabla \mathcal{F}))_+)^2 \, dx. \tag{5}$$

Thus, descent directions are obtained that prevent self-intersecting finite elements.

Following again the approach in [22], where the inner product of the shape space is modified in order to promote descent directions preventing overlaps, the term

$$\frac{\partial J_{\text{det}}}{\partial v}(v)[w_1] = -\nu_{\text{penalty}} \int_\Omega (b - \det(\nabla \mathcal{F}))_+ \operatorname{Tr}((\nabla \mathcal{F})^{-1} \nabla w_1) \det(\nabla \mathcal{F})\, dx \quad (6)$$

has to be added to the bilinear form. Note that Eq. (6) is non-differentiable but only semismooth. In order to run the semismooth Newton's method, an element of the generalized derivative has to be computed. A possible choice for a second derivative of Eq. (5) in the directions w_1 and w_2 is given by

$$\begin{aligned}\frac{\partial^2 J_{\text{det}}}{\partial v^2}(v)[w_1, w_2] = \\ \nu_{\text{penalty}} \int_\Omega \chi_{\{b > \det(\nabla \mathcal{F})\}} \operatorname{Tr}((\nabla \mathcal{F})^{-1} \nabla w_2) \operatorname{Tr}((\nabla \mathcal{F})^{-1} \nabla w_1) \det(\nabla \mathcal{F})^2 \\ + (b - \det(\nabla \mathcal{F}))_+ \operatorname{Tr}((\nabla \mathcal{F})^{-1} \nabla w_2 (\nabla \mathcal{F})^{-1} \nabla w_1) \det(\nabla \mathcal{F}) \\ - (b - \det(\nabla \mathcal{F}))_+ \operatorname{Tr}((\nabla \mathcal{F})^{-1} \nabla w_1) \operatorname{Tr}((\nabla \mathcal{F})^{-1} \nabla w_2) \det(\nabla \mathcal{F})\, dx\end{aligned} \quad (7)$$

in terms of the indicator function

$$\chi_{\{b > \det(\nabla \mathcal{F})\}} = \begin{cases} 1, & \text{if } b > \det(\nabla \mathcal{F}), \\ 0, & \text{else.} \end{cases} \quad (8)$$

The numerical results for a two-dimensional cell setting employing this non-singular displacement gradient constraint with $\nu_{\text{penalty}} = 10^2$ and $b = 5 \cdot 10^{-2}$ are presented in Fig. 6 and the behavior for the components of the objective function in Fig. 7. The same example has been chosen in [22, Fig. 4] and the computed shape positions are very similar. However, as a great benefit, significantly less optimization steps are required for the new formulation. While in [22] the final configurations has been approached with 100 steps, a very close result is now obtained within only 5 steps. As a small drawback, it turns out that the semismoothness of the modified formulation has a larger influence on the performance of Newton's method. This results in an increased number of iterations for the Newton solver and the employment of line search becomes mandatory. As a next step, we are going to investigate the scalability of this approach on the *Hawk* cluster.

4 Scalability Results

We performed weak scaling studies on the Cray XC40 *Hazel Hen* supercomputer at HLRS [13]. It features 7712 nodes with 128 GB of memory each. Every node is composed of two Intel Xeon E5-2680 processors, providing a total of 24 cores per node. In Fig. 8, we present accumulated timings and iteration counts for the first two optimization steps employing ParMETIS [10] for load balancing. The timing and speedup results are relative to 24 cores. The mesh is refined once for every core

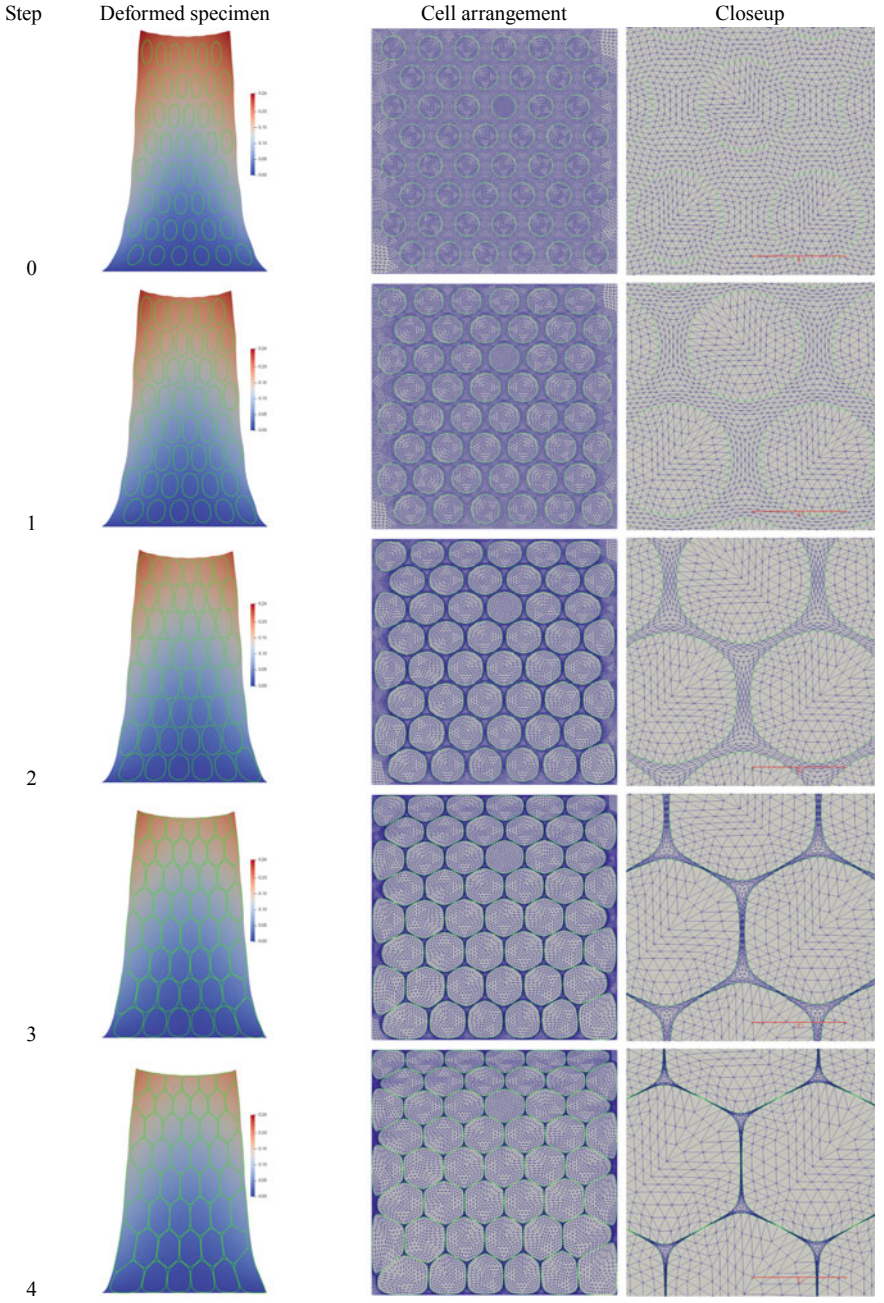


Fig. 6 Employing the constraint for a nonsingular displacement gradient with $\nu_{\text{penalty}} = 10^2$ and $b = 5 \cdot 10^{-2}$: Shape positions of the cells (green) in the deformed specimen (left) and in the FEM mesh in reference configuration (right) shown for the first 5 optimization steps

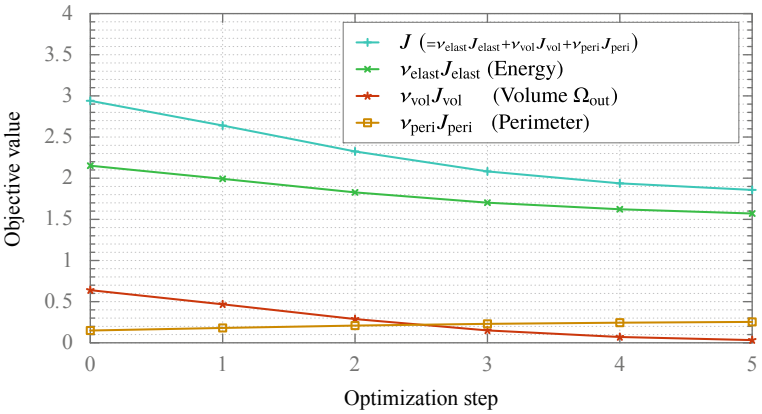
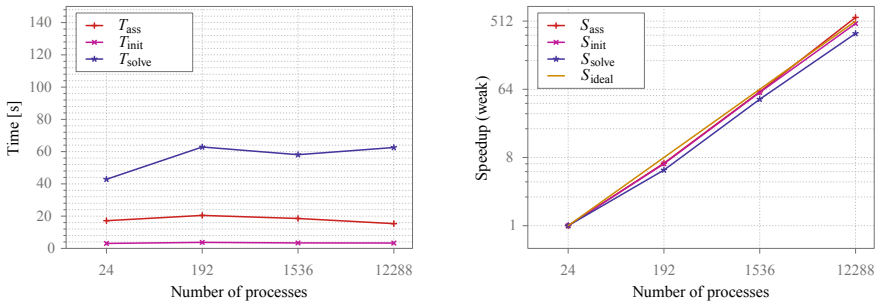


Fig. 7 Employing the constraint for a nonsingular displacement gradient with $\nu_{\text{penalty}} = 10^2$ and $b = 5 \cdot 10^{-2}$: Value of the objective function

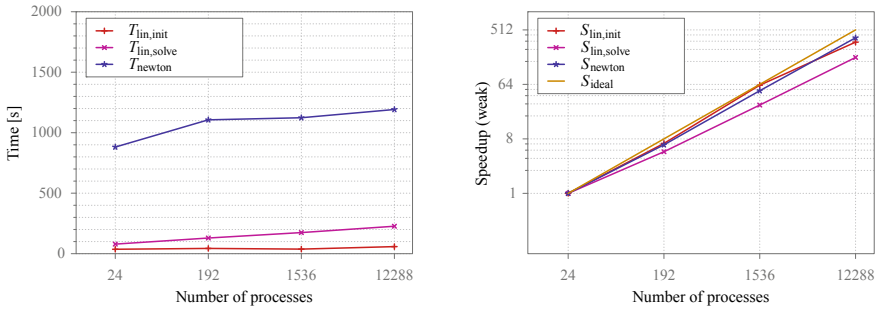
count increment, reaching more than 3 billion degrees of freedom in Fig. 8, where the maximum number of cores used is 12 288.

The shape gradient computation was performed using Newton's method, which stops at a relative error reduction of 10^{-9}. The linearized problems are solved with a BiCGStab method preconditioned by a geometric multigrid method. The multigrid uses a Block-Jacobi with 3 pre- and postsmoothing steps in a V-cycle, assembled coarse grid matrices, and a serial LU base solver on a single process. The error reduction criteria for the linear elasticity problem is 10^{-10}, and for the linearizations in the Newton method 10^{-3}, which is in this case sufficient to retain quadratic convergence.

Speedup and timings are provided for the following phases: fine level matrix assembly (ass), multigrid initialization (init), solve phases (solve) for the linear elastic problem and shape derivative, as well as for the Newton solver (newton). The results for the weak scaling are close to ideal for both the linear and nonlinear solvers. However, there is an increase in the number of steps for the linear solvers, which is attributed to the fairly different problem generated by the refinement of the mesh upon each core count increase and is reflected in an increase of the timings for the linear solver. Nonetheless, the overall results for the Newton method show a very good speedup, even for the large system of over 3 billion degrees of freedom.



(a) Linear Elastic Problem Solver



(b) Shape Derivative Solver

Procs	Refs	DoFs	Linear solver (elasticity)	Newton solver (shape derivative)	Linear solver (shape derivative)
24	4	6,085,035	159	16	293
192	5	48,530,259	186	16	382
1,536	6	387,647,139	190	17	574
12,288	7	3,098,807,619	203	19	740

(c) Accumulated iteration counts for the weak scaling study

Fig. 8 Weak Scaling (from [13]): For the first two optimization steps, the accumulated wallclock time is shown for: (a) the elastic solver, (b) the shape derivative solver. In (c), the accumulated iteration counts are presented for solving the elasticity PDE-constraint with geometric multigrid, the required Newton steps to compute the shape derivative, and the accumulated geometric multigrid steps to solve the linearization within the Newton algorithm

5 Conclusion and Outlook

We presented optimized cell arrangements that resemble the structures observed in the upper layer of the human skin. We used a linear elastic material model to simulate tensile deformations of a human skin inspired model. Starting from equally

shaped spherical shapes, PDE-constrained shape optimization has been employed to minimize for the elastic energy.

Several case studies for the weighting factors used in the optimization algorithm have been presented. These broaden our understanding of numerical shape optimization applications and provide a deeper insight into the correct configuration necessary to achieve meaningful cellular structures. We provided a weak scaling study carried out on the *Hazel Hen* supercomputer. The good scalability results are an important step towards achieving realistic epidermal cell structure simulations.

As next steps, we intend to study the scalability of the nonsingular displacement gradient approach on the *Hawk* supercomputer. In addition, our penalization scheme can be extended to other material models such as hyperelasticity.

Acknowledgment Computing time on the supercomputer *Hazel Hen* at HLRS under the grant ShapeOptCompMat (ACID 44171, Shape Optimization for 3d Composite Material Models) is gratefully acknowledged.

References

1. http://www.promesh3d.com
2. www.paraview.org
3. T.D. Allen, C.S. Potten, Significance of cell shape in tissue architecture. Nature **264**(5586), 545 (1976)
4. M. Delfour, J.P. Zolésio, Shapes and Geometries: Metrics, Analysis, Differential Calculus, and Optimization, 2nd edn., in *Advances in Design and Control*, vol. 22 (SIAM, 2001)
5. P.M. Elias, Epidermal lipids, barrier function, and desquamation. J. Investig. Dermatol. **80**(1), 44–49 (1983)
6. W. Hackbusch, *Multi-grid methods and applications*, vol. 4, (Springer, Heidelberg, 1985)
7. J. Haubner, M. Siebenborn, M. Ulbrich, A continuous perspective on shape optimization via domain transformations. SIAM J. Sci. Comput. **43**(3), A1997–A2018 (2021). https://doi.org/10.1137/20m1332050
8. H. Joodaki, B.M. Panzer, Skin mechanical modeling properties and modeling: a review. Proc. Inst. Mech. Eng. J. Eng. Med. **232**(4), 323–343 (2018)
9. J. Jor, M. Parker, A. Taberner, M. Nash, P. Nielsen, Computational and experimental characterization of skin mechanics: identifying current challenges and future directions. Syst. Biology Med. **5**(5), 539–556 (2013)
10. G. Karypis, K. Schloegel, V. Kumar, Parmetis: Parallel graph partitioning and sparse matrix ordering library (2013). http://glaros.dtc.umn.edu/gkhome/metis/parmetis/overview
11. G. Limbert, Mathematical and computational modelling of skin biophysics: a review. Proc. Math. Phys. Eng. Sci. **473**(2203), 20170257 (2017)
12. A. Nägel, M. Heisig, G. Wittum, A comparison of two-and three-dimensional models for the simulation of the permeability of human stratum corneum. Eur. J. Pharm. Biopharm. **72**(2), 332–338 (2009)
13. J. Pinzon, M. Siebenborn, A. Vogel, Parallel 3d shape optimization for cellular composites on large distributed-memory clusters. J. Adv. Simul. Sci. Eng. **7**(1), 117–135 (2020)
14. B. Querleux, (ed.) Computational Biophysics of the Skin, 1st edn. Jenny Stanford Publishing (2014). https://doi.org/10.1201/b17205
15. S. Reiter, A. Nägel, A. Vogel, G. Wittum, Massively parallel multigrid for the simulation of skin permeation on anisotropic tetrakaidecahedral cell geometries, in *High Performance Computing in Science and Engineering* 2017, ed. by W.E. Nagel et al., (Springer, Heidelberg, 2017)

16. S. Reiter, A. Vogel, I. Heppner, M. Rupp, G. Wittum, A massively parallel geometric multigrid solver on hierarchically distributed grids. Comp. Vis. Sci. **16**(4), 151–164 (2013)
17. S. Reiter, A. Vogel, A. Nägel, G. Wittum, A massively parallel multigrid method with level dependent smoothers for problems with high anisotropies. in *High Performance Computing in Science and Engineering* 2016, ed. by W.E. Nagel et al., (Springer, Heidelberg ,2016), pp. 667–675
18. S. Reiter, A. Vogel, G. Wittum, Large scale simulations of continuum models using parallel geometric multigrid methods. in *NIC Symposium* 2018 (Proceedings), ed. by K. Binder et al., (NIC Series, 2018)
19. V. Schulz, M. Siebenborn, Computational comparison of surface metrics for PDE constrained shape optimization. Comput. Methods Appl. Math. **16**(3), 485–496 (2016)
20. V. Schulz, M. Siebenborn, K. Welker, PDE constrained shape optimization as optimization on shape manifolds. in *Geometric Science of Information, Lecture Notes in Computer Science*, vol. 9389, ed. by F. Nielsen, F. Barbaresco, (2015), pp. 499–508
21. V. Schulz, M. Siebenborn, K. Welker, Efficient PDE constrained shape optimization based on Steklov-Poincaré-type metrics. SIAM J. Optim. **26**(4), 2800–2819 (2016)
22. M. Siebenborn, A. Vogel, A shape optimization algorithm for cellular composites, in *Accepted in Computing and Visualization in Science* (2019). arxiv.org/1904.03860
23. M. Siebenborn, K. Welker, Algorithmic aspects of multigrid methods for optimization in shape spaces. SIAM J. Sci. Comput. **39**(6), B1156–B1177 (2017)
24. W. Thomson, On the division of space with minimum partitional area. Philos. Mag. J. Sci. **24**(151), 503–514 (1887)
25. A. Vogel, Calotoiu, A., Strube, A., Reiter, S., Nägel, A., Wolf, F., Wittum, G.: 10,000 performance models per minute – scalability of the UG4 simulation framework. in Euro-Par 2015: *Parallel Processing*, ed. by J.L. Träff et al., (Springer, Heidelberg, 2015), pp. 519–531
26. A. Vogel, A. Calotoiu, A. Strube, S. Reiter, A. Nägel, F. Wolf, G. Wittum, Automated performance modeling UG4 simulation framework. in *Software for Exascale Computing — SPPEXA* 2013-2015, ed. by H.J. Bungartz et al., (Springer, Heidelberg, 2016)
27. A. Vogel, S. Reiter, M. Rupp, A. Nägel, G. Wittum, UG 4: a novel flexible software system for simulating PDE based models on high performance computers. Comp. Vis. Sci. **16**(4), 165–179 (2013)
28. R. Wittum, A. Naegel, M. Heisig, G. Wittum, Mathematical modelling of the viable epidermis: impact of cell shape and vertical arrangement. Math. Mech. Solids **25**, 1–14 (2017)