Welcome to Grundlsee

We are looking forward to an interesting Austrian HPC Meeting – AHPC16. This is a meeting on all aspects of scientific computing featuring keynote lectures and contributed presentations. Jointly organized by the VSC (Vienna Scientific Cluster) and ACSC (Austrian Center for Scientific Computing) it aims at bringing together scientists and technicians with a background and interest in supercomputing to discuss strategic issues for the future development of HPC in Austria. In this context a special focus will be the planning and design of the next generation of the Vienna Scientific Cluster – VSC-4. AHPC16 intends to foster new insights and collaborations for a sustainable and flourishing HPC development in Austria. We wish you an interesting time at AHPC16 in Grundlsee. If you have any questions, please feel free to contact us.

The organizing committee:

Irene Reichl (VSC)
Marián Vajtersic (ACSC)
Ernst Haunschmid (VSC)
Herbert Störi (VSC)

VSC School Projects are supported by the VSC Research Center funded by the Austrian Federal Ministry of Science, Research and Economy (bmwfw) and the therein produced computational results have been achieved using the Vienna Scientific Cluster (VSC).
<table>
<thead>
<tr>
<th>Time</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>07:30</td>
<td>breakfast</td>
</tr>
<tr>
<td>08:00</td>
<td>Ulrich Rüde</td>
</tr>
<tr>
<td>09:45</td>
<td>Christian Jordan</td>
</tr>
<tr>
<td>10:00</td>
<td>Federico Municchi</td>
</tr>
<tr>
<td>10:15</td>
<td>Lukas Einkemmer</td>
</tr>
<tr>
<td>10:30</td>
<td>Martina Prugger</td>
</tr>
<tr>
<td>10:45</td>
<td>Karl Fürlinger</td>
</tr>
<tr>
<td>11:00</td>
<td>coffee</td>
</tr>
<tr>
<td>11:30</td>
<td>Andreas Morhammer</td>
</tr>
<tr>
<td>11:45</td>
<td>Karl Rupp</td>
</tr>
<tr>
<td>12:00</td>
<td>Mathias Wiesener</td>
</tr>
<tr>
<td>12:15</td>
<td>Christian Rab</td>
</tr>
<tr>
<td>12:30</td>
<td>Francesca Nerattini</td>
</tr>
<tr>
<td>12:45</td>
<td>Claudia Stocsits</td>
</tr>
<tr>
<td>13:00</td>
<td>lunch</td>
</tr>
<tr>
<td>13:15</td>
<td>Thomas Rattei</td>
</tr>
<tr>
<td>14:45</td>
<td>Raimund Podloucky</td>
</tr>
<tr>
<td>15:00</td>
<td>Thomas Ruh</td>
</tr>
<tr>
<td>15:15</td>
<td>Lorenz Romaner</td>
</tr>
<tr>
<td>15:30</td>
<td>Oliver T. Hofmann</td>
</tr>
<tr>
<td>15:45</td>
<td>Toma Susi</td>
</tr>
<tr>
<td>16:00</td>
<td>Patrick Gunacker</td>
</tr>
<tr>
<td>16:15</td>
<td>coffee</td>
</tr>
<tr>
<td>16:45</td>
<td>Alois Schlögl</td>
</tr>
<tr>
<td>17:00</td>
<td>Samuel Senoner</td>
</tr>
<tr>
<td>17:15</td>
<td>Dietrich Liko</td>
</tr>
<tr>
<td>17:30</td>
<td>Michael Aspetsberger</td>
</tr>
<tr>
<td>18:00</td>
<td>Michael Alexander</td>
</tr>
<tr>
<td>19:00</td>
<td>discussion</td>
</tr>
<tr>
<td>19:30</td>
<td>dinner</td>
</tr>
</tbody>
</table>
Contents

Computing the heart beat – numerical challenges in solving the cardiac multiphysics problem with biophysically detailed anatomically accurate models of a human heart 1
Aurel Neic, Christoph Augustin, Anton Prassl, Andrew Crozier, Elias Karabelas, Gundolf Haase, and Gernot Plank

Heart simulation on GPU and CPU clusters 4
Gundolf Haase, Manfred Liebmann, Aurel Neic, and Gernot Plank

Eikonal equation solver on various hardware 5
Daniel Ganellari, Gundolf Haase, and Aurel Neic

Let’s talk about the weather – the role of supercomputing in earth-system modeling 6
Isabella Weger

Parametrization of atmospheric mixing processes in COSMO-CLM regional climate model 8
Andras Csaki and Heimo Truhetz

Challenges in the analysis of big earth observation data 9
Christian Briese, Werner Mücke, Richard Kidd, Thomas Mistelbauer, Stefan Hasenauer, and Wolfgang Wagner

Satellite big data processing on the Vienna Scientific Cluster for surface soil moisture estimation 10
Senmao Cao, Stefano Elefante, Vahid Naeimi, and Wolfgang Wagner

PRACE – Partnership for advanced computing in Europe 12
Thomas Ponweiser

HPC at the University of Innsbruck 13
Sabine Kreidl, and Alexander Ostermann

The Vienna Scientific Cluster 14
Herbert Störi, Peter Marksteiner, Markus Stöhr, Siegfried Höfinger, Irene Reichl, and Ernst Haunschmid

Lattice Boltzmann methods on the way to exascale 16
Ulrich Rüde

Parallel efficiency of OpenFOAM on VSC-3 18
Christian Jordan, Bahram Haddadi, and Michael Harasek

Highly efficient spatial filtering of parallel data using CPPPO: a compilation of fluid/particle post processing routines 19
Federico Municchi, Stefan Radl, and Christoph Goniva

Evaluation of the Intel Xeon Phi and NVIDIA K80 as accelerators for two-dimensional panel codes 20
Lukas Einkemmer

Evaluation of the Partitioned Global Address Space (PGAS) model for an inviscid Euler solver 21
Martina Prugger, Lukas Einkemmer, and Alexander Ostermann
DASH: data structures and algorithms with support for hierarchical locality 22
Karl Fürlinger, Tobias Fuchs, and Roger Kowalewski

Optimized sparse matrix-matrix multiplication for multi-core CPUs, GPUs, and MICs 23
Andreas Morhammer, Karl Rupp, Florian Rudolf, and Josef Weinbub

A computational scientist’s perspective on current and future hardware architectures 24
Karl Rupp and Josef Weinbub

FELTOR: a hybrid MPI + OpenMP/GPU code for plasma turbulence simulations 25
Matthias Wiesenberger, Markus Held, and Alexander Kendl

Computational astrophysics: planets, stars and galaxies 26
Christian Rab, David Bancelin, Colin Johnstone, Thomas I. Maindl, Christian Schäfer, Patrick Steyrleithner, Eduard I. Vorobyov

Introduction to the Vienna Protein Simulator: performance and applications 27
Francesca Nerattini, Luca Tubiana, and Ivan Coluzza

Molecular dynamics simulation of CD8-MHC interaction 28
Claudia Stoeckits, Reiner Ribarics, Rudolf Karch, and Wolfgang Schreiner

From Schrödinger’s equation to properties of solid matter 30
Raimund Podloucky

Evaluating the scaling behavior of WIEN2k 32
Thomas Ruh and Peter Blaha

Optimizing the mechanical properties of materials via grain boundary segregation 33
Lorenz Romaner, Daniel Scheiber, Vsevolod Razumovskiy, and Jürgen Spitaler

Computational material studies on hybrid interfaces: from structure search to property prediction 34
Oliver T. Hofmann and Egbert Zojer

Core-level calculations with hundreds of atoms using GPAW 35
Toma Susi

Diagrammatic quantum Monte Carlo for strongly correlated electron systems 36
Patrik Gunacker, Markus Wallerberger, Andreas Hausoel, Giorgio Sangiovanni, and Karsten Held

High performance computing at IST Austria: modelling the human hippocampus 37
Alois Schlögl and Stephan Stadlbauer

Responding to HPC needs in Scientific Network South Tyrol: HPC for computational linguistic purposes 38
Samuel Senoner

Evolving computing for particle physics in Austria 39
Dietrich Liko

Sentinel big data storage design and benchmarking results 40
Michael Aspetsberger, Aleksandar Djurdjic, Wolfgang Ebner, Christian Federspiel, Andreas Gerstmayr, and Martin Strigl
Parallel file I/O: using and profiling HPC storage
Michael Alexander

Challenges in geo-temporal control of multi-clouds
Ivona Brandic

Discrete load balancing in heterogeneous networks with a focus on second-order diffusion
Hoda Akbari, Petra Berenbrink, Robert Elsässer, and Dominik Kaaser

The art of benchmarking MPI libraries
Sascha Hunold, Alexandra Carpen-Amarie, and Jesper Larsson Träff

An approach to parallel nonnegative tensor factorization for HPC applications
Markus Flatz and Marián Vajteršic

Functional oxide surfaces and interfaces: DFT studies
Wernfried Mayr-Schmölzer, Florian Mittendorfer, and Josef Redinger

Algorithmic challenges in photodynamics simulations
Felix Plasser, Markus Oppel, and Leticia González

Quantum chemistry and molecular dynamics on supercomputers
Markus Oppel and Leticia González

Elucidating structural features of rigid spin crossover coordination polymers using HPC
Christian Knoll, Danny Müller, Marco Seifried, and Peter Weinberger

Highly parallel simulations of the dynamics of non-Newtonian fluids on GPUs
David Toneian, Gerhard Kahl, Gerhard Gompper, and Roland G. Winkler

Parallel software suite for neural network potentials for materials simulations
Andreas Singraber and Christoph Dellago

The million genomes challenge: bioinformatics and HPC in microbial genomics
Thomas Rattei

IQ-TREE HPC: parallelized software for phylogenetic inference
Lam Tung Nguyen, Arndt von Haeseler, and Bui Quang Minh

Ancestral and recent population genomics
Carolin Kosiol, Dominik Schrempf, Ágnes Jônás, and Christian Schlötterer

List of participants

Author index
Keynote Talk:

Computing the heart beat – numerical challenges in solving the cardiac multiphysics problem with biophysically detailed anatomically accurate models of a human heart

Aurel Neic\textsuperscript{a}, Christoph Augustin\textsuperscript{a}, Anton Prassl\textsuperscript{b}, Andrew Crozier\textsuperscript{a}, Elias Karabelas\textsuperscript{a}, Gundolf Haase\textsuperscript{b}, and Gernot Plank\textsuperscript{a}

\textsuperscript{a}Institute of Biophysics, Medical University of Graz, Graz
\textsuperscript{b}Institute for Mathematics and Scientific Computing, Karl Franzens University, Graz

Background: The heart is an electrically controlled mechanical pump, which transforms chemical energy into kinetic energy. Each beat starts with the spontaneous depolarization of cells on a timescale of milliseconds, and ends with blood flowing out of the heart to the rest of the body approximately once every second. Any major disturbance in this highly organized cascade of events may significantly impair the quality of life or even be lethal. According to the WHO cardiovascular disease (CVD) are the number 1 cause of death globally: more people die annually from CVDs than from any other cause. Considering the prevalence of cardiac disease, a better understanding of the underlying physical mechanisms is of pivotal importance and thus a major research focus.

Over the past decade, significant advances have been made in terms of experimental and clinical methodology which provide vast amounts of omics and imaging data to comprehensively characterize a patient’s cardiovascular system. However, translating this wealth of data into more effective treatments of CVD has largely proven elusive. In no small part, this can be blamed on the multiscale/multiphysics nature of cardiac function. Complex interactions of processes arise within and across multiple scales of biological organization – from genes to the organ – as well as between the multiple physics – electrophysiology, mechanical deformation and fluid flow – that orchestrate a heartbeat. This bidirectional interactions produce emergent properties which are not intuitively predictable from observing individual processes or physics in isolation. Attempting to gain a mechanistic understanding of the underlying cause-effect relationships is considered key for devising better therapeutic strategies, however, this is challenging to achieve with experimental techniques alone as these are, in general, better suited for observing single physics phenomena at one particular size scale.

Computer models are increasingly being considered an indispensable adjunct to experimental and clinical studies as a powerful quantitative framework for integrating the wealth of available data into a comprehensive representation of a patient’s cardiovascular system. Such \textit{in-silico} models of a patient’s heart allow detailed mechanistic investigations of complex cause-effect relationship across different spatial scales and physics at high spatio-temporal resolution in 3D. Currently such detailed models are being evaluated as an additional clinical modality to better inform clinical decisions by aiding in the selection of patients, optimize therapies or predict outcomes. Among the major obstacles which currently impede a wider adoption of \textit{in-silico} modeling in the clinic are, on the one hand, difficulties in parameterization and multimodal data assimilation needed to customize such models in a patient-specific fashion, and, on the other hand, the vast computational costs of executing such models. This is particularly the case for multiphysics models of total heart function where partial differential equations governing electrophysiology, deformation and fluid flow in caveties and the heart muscle itself, represented by the cardiac bidomain equation, Cauchy’s equation of motion, Navier-Stokes equation and Darcy’s equation of porous media flow, respectively, have to be solved toghether (Fig. 1).

In this talk recent methodological advances will be presented, which are suitable for simulating a heartbeat with full biophysical detail and high anatomical fidelity. Current research topics covering model building \cite{1}, parameterization and data assimilation, spatio-temporal discretization and solution techniques for the discretized systems will be covered \cite{2}.
Fig. 1: Models of total cardiac function are comprised of sub-models describing electrophysiology (Bidomain, Monodomain or Eikonal equation), structure mechanics (Cauchy's equation of motion), fluid flow in the hearts cavities and the larger attached vessels (Navier-Stokes equations), and perfusion through coronary tree and capillaries (Darcy's equation of porous media flow). Physics are bidirectionally coupled, electrophysiology and deformation through excitation-contraction coupling (ECC) and mechano-electric feedback, deformation and fluid flow/perfusion through fluid-structure interaction.

Numerical aspects and scientific computing: Models of total heart function are computationally expensive for two major reasons: i) Such models comprise multiple physics each of which is represented by a PDE; that PDE is costly to solve on its own. ii) The ongoing trend towards tomographically reconstructed, anatomically accurate models combined with biophysically highly detailed models of cellular dynamics increased the overall degrees of the problem and as such the computational demands substantially. For instance, electrical activation of a myocyte is driven by a very fast depolarization event where a cell’s state changes from rest to activation within a millisecond. These fast transients in time translate into steep wavefronts in space which imposes severe restrictions on the spatio-temporal discretization. Typically, time steps $dt < 25 \mu s$ and spatial discretizations $h$ in the range between $50 \mu m$ up to $300 \mu m$ are used. This translates into a system of equations of around $10^9$ - $10^{10}$ million degrees of freedom which has to be solved between $10^4$ - $10^6$ times to compute one heartbeat. In addition, a system of ODEs comprising between $10^2$ - $10^6$ non-linear ODEs has to be solved at every point in space to compute the current state of cellular dynamics.

Two alternative approaches are being pursued to address this issue. One approach relies upon spatio-temporal adaptivity, which aims at reducing the dimensionality of the problem by using high spatio-temporal resolutions only around steep wave fronts and during fast transients. The alternative strategy follows the more hardware driven many-core computing approach where a reduction in execution times is achieved by employing hundreds to thousands of compute cores, be it traditional CPUs or, more recently, accelerators such as GPUs or Intels Phi. Both approaches have their pros and cons. Spatio-temporal adaptivity provides better accuracy and may allow executing organ-scale simulations in less powerful computing environments. However, the current trends in HPC hardware clearly favor parallelization approaches. Achieving good strong scaling characteristics with fixed grid codes is challenging, but feasible with less resources relative to spatio-temporally adaptive codes where major challenges have to be addressed such as devising and implementing efficient dynamic parallel load balancing or minimizing the costs of error estimation and mesh refinement. Currently, the parallelization approach seems to have a distinct advantage over adaptive
methods in terms of performance as well as general applicability, although noticeable progress has been made also with both h- and p-adaptivity. Addressing scalability is further complicated by the current exa-scale trend towards extremely large core counts, which require further reductions in communication to fully exploit the hardware.

Computing a heartbeat: In a recent study of ours we reported on the development of novel methodology for solving the nonlinear equation of finite elasticity using human whole organ models of cardiac electromechanics, discretized at a high para-cellular resolution [2]. A patient-specific, anatomically accurate, whole heart electro-mechanical model was reconstructed from magnetic resonance (MR) scans (Fig. 2A) at three resolutions of 220µm, 440µm and 880µm, yielding meshes of approximately 184.6, 24.4 and 3.7 million tetrahedral elements and 95.9, 13.2 and 2.1 million displacement DOF, respectively. The same mesh was used for discretizing the governing equations of both electrophysiology and nonlinear elasticity. Fig. 2B shows electrical wavefront propagation activating the heart. A novel algebraic multigrid (AMG) preconditioner for an iterative Krylov solver was developed to deal with the resulting computational load. The AMG preconditioner was designed under the primary objective of achieving favorable strong scaling characteristics for both setup and solution runtimes, as this is key for exploiting current high performance computing hardware. Benchmark results using the 220µm, 440µm and 880µm meshes demonstrate efficient scaling up to 1024, 4096 and 8192 compute cores which allowed the simulation of a single heart beat in 44.3, 87.8 and 235.3 minutes, respectively (see Fig. 2). The efficiency of the method allows fast simulation cycles without compromising anatomical or biophysical detail.

Conclusions: Modeling of total cardiac function using anatomically accurate and biophysically detailed models of an individual patient’s cardiovascular systems is becoming feasible. In the quest of maturing such models towards routine clinical applicability, the implementation of numerical methods which make efficient use of massively parallel HPC resources and/or the exploitation of acceleration technologies is playing a pivotal role.

References
Heart simulation on GPU and CPU clusters

Gundolf Haase\textsuperscript{a}, Manfred Liebmann\textsuperscript{a}, Aurel Neic\textsuperscript{b}, and Gernot Plank\textsuperscript{b}

\textsuperscript{a}Institute for Mathematics and Scientific Computing, Karl Franzens University of Graz
\textsuperscript{b}Institute of Biophysics, Medical University of Graz

We developed our algebraic multigrid solvers (AMG) and preconditioners with special focus on cardiac electromechanics with the goal of simulating one heart beat as fast as possible. The overall systems consist of the bidomain equations (elliptic + parabolic partial differential equations (PDE), non-linear coupling via an ordinary differential equation (ODE) system) \cite{2,3} coupled with non-linear elasticity for large deformations. The simulation is based on unstructured 3D meshes with anisotropic, inhomogeneous material coefficients.

Besides choosing the AMG components such that the overall runtime is minimized, we needed a highly efficient MPI + OpenMP parallelization with an additional acceleration on GPUs. The presentation will focus on all the little improvements necessary to achieve very good strong speedup on 4096 CPU cores \cite{1} such that one coupled system with 7 Mill. degrees of freedom can be solved in less than one second. One large parallel performance improvement has been achieved by using an extra balanced partitioning for data on subdomain interfaces. As a consequence the parallel strong scaling of the conjugate gradients solver with the best AMG preconditioner (hybrid MPI+OpenMP) is as good as using the much simpler Jacobi precondition that has a 10 times longer run time, see Fig. 1. The number of cores is reduced on coarser grids to reduce the communication overhead.

The parallelization for many-core processors as NVIDIA GPUs results in further speedup between 5 and 30 depending on the subtasks. In order to reduce the data transfer between accelerator memory and CPU memory in the non-linear solvers, we had to redesign the interfaces and data structures in the whole medical simulation code according to plain data structures and flexible solver steps. Although this parallelization has been done in CUDA \cite{2} the future development will use pragma driven parallelization in OpenACC (GPU) and/or OpenMP 4.0 (Intel Xeon Phi) in order to have one code for all current (and future) many-core hardware.

Supported by the FWF project F32-N18 and by NAWI Graz.

References

Eikonal equation solver on various hardware

Daniel Ganellari\textsuperscript{a}, Gundolf Haase\textsuperscript{a}, and Aurel Neic\textsuperscript{b}

\textsuperscript{a}Institute for Mathematics and Scientific Computing, Karl Franzens University of Graz
\textsuperscript{b}Institute of Biophysics, Medical University of Graz

Simulations of one heart beat which faithfully account for biophysical details involved in cardiac electrophysiology and mechanics are still far away from real time performance, even when employing several thousands of compute nodes. This talk presents a simpler model based on the Eikonal equation and its numerical solving.

The non-linear Eikonal equation

\[
\sqrt{(\nabla \varphi(x))^T M(x) \nabla \varphi(x)} = 1 \quad x \in \Omega
\]

describes a traveling wave through the domain $\Omega$ with given heterogeneous, anisotropic velocity information $M$. The solution $\varphi(x)$ denotes the time when the wave arrives at point $x$.

We discretize the domain by planar-sided tetrahedralization. Based upon this tetrahedralization, we form a piecewise linear approximation of the solution by maintaining the values of the approximation and employing linear interpolation within each tetrahedral element in the discretized domain. To accomplish this we use the fast iterative method proposed initially at \[1\] and extended in \[3\] for tetrahedral meshes. We improved the algorithm, implemented it on various hardware with very good results and also transferred the solver for a coarse model to a tablet computer and other hand-held devices for potential clinical use.

The shared memory parallelization on CPU has been implemented using OpenMP and we are currently working to finish the shared memory parallelization on GPU using CUDA. This implementation will reduce the run time further such that also interactive simulations will be possible. Due to the splitting of the wave front, the parallel version results in a slightly different convergence history and in minor differences in the solution. The parallel algorithms have been tested on workstations and on Android devices. We have a very short convergence time of the algorithm and good quality results, see Fig. 1 wherein the wave propagation looks very smooth. Remaining staircase effects caused by obtuse tetrahedra have been reduced by the technique proposed in \[3\].

\textbf{Fig. 1:} Arrival time $\varphi(x)$ ranging from 0 (bottom) to 1 (top).

Supported by JoinEU-SEE PENTA scholarship scheme, partially supported by the FWF project F32-N18 and by NAWI Graz.

References

Keynote Talk:

Let’s talk about the weather – the role of supercomputing in earth-system modeling

Isabella Weger

European Centre for Medium-Range Weather Forecasts (ECMWF), Reading, UK

Weather influences the lives of everyone. Society requires more accurate and reliable information on weather and climate for a wide range of endeavors. Early warnings of severe or high-impact weather, such as windstorms, floods or heat-waves, are crucial to protect live and property. There are numerous uses and users of weather forecasts in our daily lives, in agriculture, transportation, energy, financial and insurance sectors and many more. Monitoring the climate and understanding the impacts of a changing climate is becoming ever more important for society.

For centuries, people have tried to predict the weather, often relying on observing current weather conditions and patterns to give guidance whether the next day will bring fair or foul weather. Numerical weather prediction (NWP) uses the laws of physics that govern the behaviours of atmosphere and oceans. The fundamental physical equations have been formulated at the beginning of the 20th century, but their implementation for the calculation of forecasts needed the computing technology that became available in the 1950s. Since then, the numerical methods and model accuracy have been continually improved and taken advantage of the rapid evolution of computing and microprocessor technology.

NWP today uses sophisticated numerical models to simulate the evolution of the complex processes in the atmosphere and ocean. Starting point is the information about the current state of the weather. Meteorological observations, gathered from weather stations, ships, aircraft, balloons, satellites etc, are analyzed and fed into a physical model of the atmosphere, a process known as data assimilation. The forecast models solve numerically the equations that describe the changes of wind, pressure and temperature in space and time.

Today, operational NWP centres worldwide produce a wide range of forecasts, from the very short range, i.e. hours or days ahead, to weeks and months and up to global seasonal forecasts. ECMWF, for example, produces global 15-day forecasts twice per day. More than 40 million observations per day are used to derive the initial state of the atmosphere. With a 16 km horizontal resolution and 137 levels in the vertical, the model forecasts wind, temperature and humidity at nearly 300 million grid points throughout the atmosphere at each time step. In addition, an “ensemble prediction”, a set of 51 forecasts from perturbed initial conditions provides an estimate of forecast uncertainty. ECMWF also produces monthly and seasonal predictions and “reanalysis” combining past meteorological observations into comprehensive atmospheric data sets for climate monitoring and research.

Weather forecasting as we know it today would not be possible without the use of state-of-the-art supercomputers. ECMWF’s first operational forecast in 1979 was run at 210 km global resolution on a single-processor Cray-1A. ECMWF’s current high-performance computing facility (HPCF) [1], two Cray XC-30 clusters with a total of over 160,000 processor cores, runs a 16 km model today, and in spring 2016, a forecast model with 9 km grid resolution (and more advanced science) will be introduced into operation. ECMWF’s Integrated Forecasting System (IFS) reflects leading scientific developments in numerical weather prediction over more than 30 years in an application with more than two million lines of code. Over time, various HPC architectures have been used, including CRAY vector shared-memory systems, Fujitsu VPP vector distributed-memory systems, IBM scalar SMP clusters, and most recently, Cray x86-based supercomputers. Portability across the prevailing HPC architectures has always been important. Supercomputers are at the core of the production process of weather forecasts. NWP centres are expected to run weather forecast models within a reasonably short times lot, and to deliver a consistent and reliable service each and every day. The goal is not only to produce the best possible forecasts, but also to deliver them on time to a
strict production schedule. This requires a resilient configuration of the HPCF, robust 24/7 operations and appropriate scheduling of operational and research workloads.

The success story of numerical weather prediction has been described as a “quiet revolution” [2], fuelled by steady scientific progress and technological advances. Like for many other fields involving computational sciences, computing has been an enabling technology for numerical weather prediction and climate modeling. The growth of computational performance allowed for example increasing model grid resolution and model accuracy, better representation of physical processes and advanced handling of observations to produce the better initial conditions for forecasts. Forecast skill improvement has followed and enabled the production of good forecasts of weather into the medium range and beyond. For the forecast range of 3 to 10 days ahead, forecast skill has increased by about one day per decade — today’s 6-day forecast is as accurate as a 5-day forecast from ten years ago. Advances in HPC technology and performance also allow more comprehensive earth-system models to be developed: the atmospheric model is coupled to other models, such as ocean, wave, sea ice and land-surface models. Including the composition of the atmosphere (e.g. aerosols and greenhouse gases) in the models enables new applications such as the prediction of air quality.

For future generations of earth-system models significant increases in both model complexity and model resolution are envisaged. One of the key challenges is how to make use of increasing computing performance with significantly better energy-efficiency on future supercomputer systems. Current HPC developments are towards massive core counts, accelerators, many-core processors, and CPU-GPU heterogeneous computing [3]. HPC technology developments are influencing the directions that NWP will take, requiring paradigm changes regarding numerical methods and the design of codes. Exploiting parallelism on all architectural levels and improving the scalability of all codes will be crucial for the progress in numerical weather prediction and climate modeling.

References


Parametrization of atmospheric mixing processes in COSMO-CLM regional climate model

Andras Csaki and Heimo Truhetz

Wegener Center for Climate and Global Change, University of Graz

In regional climate models, the mixing of momentum and scalar quantities in the planetary boundary layer highly depends on the representation of turbulence and convection. In the so-called convection permitting simulations, deep cumulus convection is assumed to be resolved (at least partially) and a deep-convection parameterization scheme is switched off. Shallow convection is still to be parametrized, because this process remains at sub-grid scales (SGS). In the regional climate model COSMO-CLM, two turbulence parameterizations schemes can be used: (i) the default TKE scheme, and (ii) the TKESV (Turbulent Kinetic Energy Scalar Variances) scheme recently developed at Deutscher Wetterdienst (DWD) [1]. In the default COSMO-CLM configuration, the TKE scheme is used to describe the small-scale chaotic turbulence, and a shallow convection scheme is used to describe quasi-organized (still SGS) cumulus-type structures. It is expected that the new TKESV scheme is capable of describing both turbulence and shallow convection in a unified framework. Then, the COSMO-model shallow-convection scheme may be switched off. The present study addresses two issues: (i) what is the relation between deep and shallow convection in COSMO-CLM, and (ii) would the TKESV scheme be capable of adequately describing shallow convection.

Simulations are performed with the COSMO version 5.0, using the TKE and TKESV turbulence schemes and the shallow-convection scheme on and off. These sensitivity experiments are run with 3 km horizontal mesh size (“convection permitting” resolution) over the European Alpine region. The boundary conditions for the regional model are taken from the Integrated Forecast System (IFS) of the European Centre for Medium-Range Weather Forecasts (ECMWF) [2]. Results from four simulations are compared to each other and to the results from ECMWF IFS, from COSMO-7 [4] of MeteoSwiss, and (for some fields) from the Austrian nowcasting system (INCA) [3]. A large number of meteorological fields are analyzed, including 2-metre temperature, precipitation, cloud cover, and mean vertical velocity and its variance.

The work is performed within the framework of the project ”Non-Hydrostatic Climate Modelling II” (NHCM-2; www.nhcm-2.eu) funded by the Austrian Science Fund (FWF; project ID P 24758-N29). The calculations have been performed using the resources of the VSC.

References


Challenges in the analysis of big earth observation data

Christian Briese\textsuperscript{a}, Werner Mücke\textsuperscript{a}, Richard Kidd\textsuperscript{a}, Thomas Mistelbauer\textsuperscript{a,b}, Stefan Hasenauer\textsuperscript{a}, and Wolfgang Wagner\textsuperscript{b,a}

\textsuperscript{a}Earth Observation Data Centre for Water Resources Monitoring (EODC)

\textsuperscript{b}Department of Geodesy and Geoinformation, TU Wien

Within the European earth observation (EO) programme Copernicus a huge amount of Sentinel satellites providing data for environmental protection, urban planning, agriculture, forestry and water economy, and civil protection will be launched till 2030. Due to the advanced sensing concepts and outstanding spatio-temporal sampling characteristics, the Sentinels will collect more and better data than ever before. Exploiting their data is challenging due to (1) storing, transferring and processing such big EO data, and (2) the lack of adequate processing algorithms, optimised and exploited jointly by different disciplines.

The Earth Observation Data Centre for Water Resources Monitoring (EODC) was founded in 2014 as private-public partnership to tackle the above mentioned challenges and to enable the effective and efficient usage of big EO data. Furthermore, EODC facilitates the cooperation among the scientific, public and private sectors. The concept of EODC’s IT infrastructure and framework brings scientists, developers and data analysts together on a single platform, thereby fostering productive and collaborative working methods.

This contribution will give an introduction to EODC’s virtual research, development and operations environment that consists of three key pillars: (1) the Science Integration and Development Platform (SIDP), which is a fully equipped cloud computing infrastructure, (2) the Near Real-time Operations and Rolling Archive (NORA), which is a high-availability storage and processing cluster, and (3) the Global Testing and Reprocessing Facility (GTR), based on the Vienna Scientific Cluster 3 (VSC-3) connected to EODC’s big EO data storage. The GTR serves to run algorithms on global scale or reprocess complete historic data archives on a regular basis. It is demonstrated with practical examples how these IT capacities can be employed for collaborative methods and software development and testing, as well as accessing and processing a Petabyte-scale archive of EO data.

The complete suite of resources underpinning the EODC framework are accessible via a central and browser-based web portal and interactive delivery platform, which provides the necessary tools for building, testing, (near real-time) delivery and analysis of value-added EO products. By providing an environment that connects science, development and operations the EODC is a catalyser for open and international cooperation amongst public and private organisations and fosters the use of EO for local and global monitoring tasks.

References

Satellite big data processing on the Vienna Scientific Cluster for surface soil moisture estimation

Senmao Cao, Stefano Elefante, Vahid Naeimi, and Wolfgang Wagner

Department of Geodesy and Geoinformation, TU Wien

Earth observation satellites such as Envisat and Sentinel-1 are producing huge amounts of data, which poses new challenges to the worldwide scientific community[1, 2]. These satellite data are very useful to provide a surface soil moisture data to both operational users and scientific community [3]. With normal or high performance PC, it is impossible to process within a reasonable time span. Therefore, high computing capability for processing the huge volumes of satellite data is needed. The Vienna Scientific Cluster 3 (VSC-3) installed in summer 2014 with 2020 nodes has been designed to meet the requirements for high performance computing of a consortium of Austrian universities. It makes the processing of such big data within the certain time frame possible by exploiting the parallel processing.

Different preliminary tests have been conducted on VSC-3 to investigate the feasibility to process the whole Envisat Advanced Synthetic Aperture Radar (ASAR) Global Mode (GM) archive and Sentinel-1 Ground Range Detected (GRD) high resolution data acquired from October 2014 to September 2015. An Envisat ASAR GM dataset consisted of 84449 acquisitions about 550 GB in total has been processed with an array of 20 jobs. However, due to the I/O characteristics of our code which accesses a large number of small files performance penalties could be detected. Further investigation is ongoing to try to find a better implementation that can solve the problems of intensive I/O operations. Nonetheless, when using only 20 nodes, the performance of the processing was still satisfactory. Two different sets of Sentinel-1 images, covering two geographical regions constituted by 624 and 879 images respectively, have been launched on VSC-3. On each node, due to the RAM limitation two different images were simultaneously processed and therefore 312 and 435 jobs array were requested. The average processing time has been 45 and 120 minutes respectively, the difference has been caused by the physical characteristics of the two considered geographical regions. The time needed to transfer the data through the network and to write on disk has been satisfactory when compared to the CPU time.

The performed experiments showed that VSC-3 can be used for processing satellite big data such as Envisat and Sentinel-1 to estimate surface soil moisture. Further investigation is needed to optimise the processing on the VSC-3.

References


PRACE – Partnership for advanced computing in Europe

Thomas Ponweiser

RISC Software GmbH

The EU project PRACE (Partnership for Advanced Computing in Europe, www.prace-ri.eu) establishes a pan-European research infrastructure, providing access to supercomputing resources and services for large-scale scientific and engineering applications. Among the currently 25 member states, Austria is represented by the Research Institute for Symbolic Computation (RISC), an institute of the Johannes Kepler University Linz (JKU), and RISC Software GmbH (RISCSW; as a third party associated to RISC).

This talk will provide general information on PRACE and its services (e.g. how to apply for access to PRACE HPC resources; or also the PRACE Autumn School 2016, taking place in Hagenberg, Upper Austria, will be advertised). The main focus, however, will lie on our contributions and achievements within PRACE so far, in particular in the field of providing support for European domain scientists in profiling, tuning and scaling scientific high performance applications.

In the frame of PRACE Preparatory access, the performance and scalability of the quantum mechanics code wannier90 [1] and the astrophysics code Parsek2D-MLMD [2] have been improved significantly. The speedups for typical computations with these codes range between factors of 5 to 10 and good scalability up to 2048 processes has been demonstrated in both cases.

Moreover, in the frame of PRACE Socio-economic Challenges, we implemented a prototype for a high-fidelity coupled fluid dynamics / structural mechanics simulation based on the open-source community code OpenFOAM. As a side-effect of our work, we were able to identify and eliminate a severe scalability bottleneck in OpenFOAM’s inter-processor communication scheme, which increased the scalability of our prototype from below 512 to above 4096 processes for a model with 61 million cells [3].

References


HPC at the University of Innsbruck

Sabine Kreidl\textsuperscript{a} and Alexander Ostermann\textsuperscript{b,c}

\textsuperscript{a}Zentraler Informatikdienst, University of Innsbruck
\textsuperscript{b}Focal Point Scientific Computing, University of Innsbruck
\textsuperscript{c}Department of Mathematics, University of Innsbruck

The focal point Scientific Computing [1] is one of the five major research areas of the University of Innsbruck. It integrates all research activities of the university in the field of high performance computing. Together with the IT Center of the university, it runs two distributed memory infiniband clusters, LEO3 and LEO3E. Moreover, it operates the shared memory system MACH together with the University of Linz, and it is member of the Vienna Scientific Cluster VSC-3. The purpose of this talk is to give a brief overview of the scientific activities of the focal point, of its local infrastructure, and of the ACSC, the Austrian Center for Scientific Computing.

With its High Performance Computing (HPC) department, the university’s IT Center has a special focus on the needs and challenges of scientific computing users. For them not only the sufficient supply of computational resources is of major importance. Easy access, ease of use and professional support are just as relevant. By providing technical guidance, an extensive software portfolio, appropriate tools and a uniform look and feel, highly efficient use of the systems is facilitated. And the entry threshold for new users is significantly lowered. This talk will also provide a cursory insight into the facilities and mechanisms employed for this purpose at the University of Innsbruck.

References

The Vienna Scientific Cluster

Herbert Störi, Peter Marksteiner, Markus Stöhr, Irene Reichl, Siegfried Höfinger, and Ernst Haunschmid

VSC Team

Introduction (Herbert Störi)

The Vienna Scientific Cluster (VSC) is a joint high performance computing (HPC) installation and service of Austrian universities. The original VSC, later termed “VSC-1” was an initiative by the University of Vienna, the University of Natural Resources and Life Sciences, Vienna, and the TU Wien. Now also the Graz University of Technology (together with the universities within “Universitäts-Cluster-Süd”) and the University of Innsbruck are members of the consortium. As technology evolves rapidly, we are regularly procuring new VSC’s. Presently VSC-1 is largely decommissioned, VSC-2 and VSC-3 are in full operation. At this meeting we will discuss the results obtained, but also the requirements for a new cluster, VSC-4, to be tendered later this year. As the VSC is a facility driven by user demand, this discussion is a very important input at the start of the VSC-4 procurement.

The VSC-3 technology stack (Peter Marksteiner)

A short presentation of the entire technology stack of the VSC-3 supercomputer is given, in order to give an idea of the “inner workings” of a supercomputer usually hidden from the end user. This stack comprises, among others: basic infrastructure and cooling, hardware components like nodes, storage servers, switches and cabling; network fabric, management, monitoring, and deployment tools, security concepts, batch system, parallel file systems, operating systems, compilers, libraries and debuggers, and HPC applications. Some of these components, like the innovative oil-based cooling system and the “dual rail” InfiniBand fabric are highlighted and presented in detail. Several challenges are described that had to be overcome to achieve stable and productive operation of the VSC-3.

VSC-3 status and best practices (Markus Stöhr)

This talk gives an overview of the status of VSC-3 and gives some hints on how to use it in an efficient way. Covered topics are:

- Login procedure with one time password (OTP) and the security considerations behind it.
- Overview of the current configuration of the batch system SLURM.
- Hints on the module environment.
- Guidelines on how to report problems to user support.

GPU and MIC computing on VSC-3 (Siegfried Höfinger)

Basic operation of accelerator cards is described within the VSC-3 environment and corresponding benchmarks are presented. Potential pitfalls are identified and practical recommendations given to allow smooth initiation of user activity on VSC-3 [1].
Particular attention will be paid to the HPL benchmark [2] and its ported version to the GPU [3]. Raw HPL performance data obtained on GPUs of type C2050 (fermi) and K20m (kepler) are shown in Figure 1. Various technical aspects will be addressed and the general subject of accelerator based HPC will be put into a broader perspective. A shortlist of important considerations for future strategic planning will be provided for discussion.

Fig. 1: HPL performance in TFLOPs obtained on GPUs of type C2050 (fermi) and K20m (kepler). Numbers above the bars indicate accelerations with respect to a single GPU. General HPL performance obtained with 5 standard nodes on VSC-3 is also included for comparison (red dotted line).

Remote visualization on VSC-3 (Irene Reichl)

Remote data analysis and visualization become necessary when the amount of data exceeds the hardware capabilities of the local desktop or if repeated data transfer of new results would take too long. In that case, a Virtual Network Computing (VNC) client transmits the keyboard and mouse events from the desktop device to the remote render server where VirtualGL and GLX send Open Graphics Library (OpenGL) commands directly to the hardware providing accelerated rendering. In addition, VNC allows another user to access the same session in view-only mode, facilitating the discussion of research results with partners. A step-to-step usage guide of VNC and VirtualGL can be found in the VSC Wiki: GPU computing and visualization, https://wiki.vsc.ac.at/doku.php?id=doku:vsc3_gpu

On VSC-3, available programs for visualization include Vtk, Paraview, Matlab, and Mathematica. Presently, a test-installation on VSC-3 provides five nodes equipped with GPU devices that can be used for remote visualization. Software and resources are to be customized according to user demand.

Architectures for future HPC systems (Ernst Haunschmid)

An overview of current and future architectures and technologies for HPC is given. There is a strong trend towards integration of accelerators (GPUs, Xeon Phi, ...) into traditional HPC systems. Advantages and disadvantages of a few widely used designs will be discussed. On the system level OpenPower has become an attractive alternative to Intel-based systems, especially in combination with Nvidia next generation Volta GPUs. This overview is intended as an introduction to a discussion about needs, demands, and requirements from VSC customers for a next generation VSC system.

References


**Keynote Talk:**

**Lattice Boltzmann methods on the way to exascale**

Ulrich Rüde  
*Lehrstuhl für Simulation, Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany*

In the past decade, the lattice Boltzmann method (LBM) has gained popularity as an alternative to classical Navier-Stokes solvers for computational fluid dynamics (CFD). With the LBM, the simulation domain is discretized with a uniform Cartesian grid. If the resolution of a three-dimensional simulation must be increased in space and time, then the total number of cells and the computational cost increase quickly. However, as an explicit scheme, the LBM requires only a data exchange between nearest neighbors in each time step.

Many implementations of the LBM are therefore designed for parallel computers. Going beyond scalability alone, a carefully crafted, architecture-aware implementation of the LBM, as realized in the waLBerla framework [3] can achieve excellent absolute performance and thus reduce the time to solution to reach a given computational objective. This requires a specific tuning and optimization of the computational kernels to exploit modern hardware features such as vectorization with SIMD units. We will report on simulations using the D3Q19 (3 dimensional 19 velocity lattice) model and the two-relaxation-time (TRT) model. For this scenario, we will show that the extra numerical work required for the TRT scheme does not cost additional computing time when the computational bottleneck happens to be the memory bandwidth on each node rather than floating point performance. As in many bandwidth-limited codes, this situation is only reached after the memory access structure has been analyzed carefully and is then optimized manually. Unfortunately, even modern optimizing compilers still need substantial help by programmers to generate efficient code and of course they rely on suitably designed data structures that do not create other computational bottlenecks.

As a result of the performance-aware co-design, the waLBerla framework can discretize a complex flow geometry with in excess of a trillion ($10^{12}$) lattice cells on current petascale supercomputers with up to 458,752 processor cores [3]. Note that many parallel LBM codes can scale to such core numbers, but only few will reach the absolute performance that is reported here since this relies on using the concurrency within each node and in each core, combined with the scalability across nodes.

Further algorithmic extensions are necessary to permit adaptive mesh refinement and handling free surfaces. For these more complex flow scenarios, the parallelization becomes much more complicated and they become more compute intensive. In particular, we will report on coupled simulations with particles embedded in the fluid. To limit the computational effort, many previous simulations were performed either with only few particles or in only two dimensions, or with simplified models for the fluid-particle interaction. Our new simulation method and its implementation extend these approaches and allows for fluid simulations in 3D with millions of interacting particles [2] suspended in the flow. For this, we rely on algorithms for rigid body dynamics, where particles are modeled as individual geometric entities in a Lagrangian setting. Each particle is characterized by its coordinates and velocity including its rotational degrees of freedom. Particles interact through exchanging impulses when they are in contact. The frictional multi-body contact problem is formulated with measure differential inclusions that lead to nonlinear complementarity problems in each time step. The coupling with the hydrodynamics is achieved via the momentum exchange method and imposing the correct boundary conditions for the flow on the particle surfaces. Our simulation framework includes the so-called lubrication correction for suspended particles and can be extended to include further effects, such as electrostatic forces. A thorough analysis with performance models demonstrates that these coupled simulations still achieve excellent computational performance.
In recent years, computing with graphics processors (GPUs) has received widespread interest. In the light of the above discussion on node performance and intra-node code optimization, a GPU version of the LBM [1] can typically provide speedups of around a factor 3-5 compared to CPU codes on hardware of the same technology generation. GPU speedups of a factor of 100 or more that are sometimes reported in the literature cannot be substantiated. Such speedups from CPU to GPU are only observed when the programs that are used for comparison do not exploit the given architecture effectively, e.g. when the memory bandwidth of a given system cannot be exhausted by using data structures that lead to unfavorable algorithmic memory access patterns.

An increasing number of real-life applications illustrates the generality and the power of the parallel LBM approach. These include the development of closure relations for macroscopic multiphase models, the study of self-propelled swimmers, processing metal foams, patient specific blood flow, food technology, and additive manufacturing.

References


Parallel efficiency of OpenFOAM on VSC-3

Christian Jordan, Bahram Haddadi, and Michael Harasek

Institute of Chemical Engineering, TU Wien

Computational fluid dynamics (CFD) has proven to be a valuable tool for the investigation of physical and chemical phenomena in chemical engineering applications. One of this applications from the field of thermal process engineering is adsorption [1]. Adsorption is a process in which molecules from a fluid phase (gas or liquid) are selectively bound to a solid phase, the adsorbent. In many cases the adsorbent consists of granular particles which are filled into a reactor as a random packed bed. The local flow field, temperature distribution and the adsorbed amount in such a packed bed are of great interest for chemical engineers to improve and to optimize this processes. However, CFD requires sufficient spatial resolution of the geometry investigated – finite volume discretization of a packed bed of particles can result in computational grids of 10–100 million cells. The solution of such CFD cases can only be done using free software, e.g. the open source tool OpenFOAM. Also adequate hardware infrastructure needs to be provided, e.g. the HPC VSC-3.

OpenFOAM [2] provides multiple tools for splitting meshes into portions (partitioning) and performing MPI parallelization of CFD cases. In earlier test runs it was found that not only the case, the solver and the hardware configuration have impact on the parallel performance but there is also a significant influence of the method used for mesh partitioning. For a more detailed analysis a practical example of a packed bed adsorber filled with spheres was selected. The geometry was created using DPMfoam for preparing the packed bed consisting of about 2000 spheres. The packing structure was mapped into a sufficiently fine hexahedral base mesh resulting in about 60 million cells in two regions. Gas flow and adsorption was solved using adsorpFoam, a customized solver derived from reactingFoam and chtMultiRegionFoam. An example of the flow field is shown in Figure 1. Decomposition of the mesh was carried out using “simple”, “hierarchical” and “scotch” in multiple ways resulting in 8–2048 partitions.

To compare the various mesh decomposition variants flow calculations were carried out for a fixed period of time (0,5 s physical time) and the time consumption was recorded. From this data the parallel efficiency was calculated. From Figure 2 it is obvious that the method “simple” has the most inefficient partitioning and therefore high communication losses. The fully automatic method “scotch” gives average performance (but has high one-time initialisation and loading effort). The best speedup can be achieved using the more manual method “hierarchical” where splits are carried out in main axis direction first (zxy) - other splitting sequences can lead to lower performance (see e.g. xyz).

It is also clear that above ∼ 512 cores the overall speedup and efficiency are quite inferior – decomposition with too high core numbers (resulting in <100000 cells per core) has to be avoided.

References

Highly efficient spatial filtering of parallel data using CPPPO: a compilation of fluid/particle post processing routines

Federico Municchia, Stefan Radla, and Christoph Gonivab

aInstitute for process and particle engineering, TU Graz
bDCS Computing GmbH, Linz

Multi-scale approaches are widely used to model phenomena in physics and engineering disciplines. This is due to the broad range of temporal and spatial scales at which relevant phenomena take place [1]. Such a multi-scale approach typically relies on the use of data from fully resolved simulations to derive closures for models on larger scales. These closures are often difficult to establish [2-3], and often require the processing of a huge amount of data from different time steps.

In this work, we present CPPPO: a universal open-source library for developing such closure models by filtering data from resolved simulations.

CPPPO features several parallel filtering algorithms and sampling utilities designed to process data "on-the-fly" from massively parallel numerical simulations. This library is linked to the widely-used finite volume library OpenFOAM®. In addition, it is provided with a general interface to read csv data files. Furthermore, the library can be used to process data from both structured and unstructured grids. CPPPO requires a minimum amount of coding, and its modular structure allows the user to easily introduce new filtering kernels or new sampling operations.

CPPPO was proven to be accurate and fast. Results from scalability analysis conducted using the Vienna Scientific Cluster 3 (VSC-3) showed values of the strong parallel efficiency much higher than 100%. In fact, the library introduces a novel algorithm for parallel filtering that minimizes the number of communications between processors, leading to an outstanding parallel performance.

CPPPO was shown to require just a few per cent of the total computational time when used to process data from large parallel simulations of fluid flow in dense gas-particle systems.

References
Evaluation of the Intel Xeon Phi and NVIDIA K80 as accelerators for two-dimensional panel codes

Lukas Einkemmer

Department of Mathematics, University of Innsbruck

To predict the properties of fluid flow over a solid geometry is an important engineering problem. Numerical simulations are routinely used in applications ranging from the design and analysis of aircrafts to constructing more efficient wind turbines. If the flow under consideration is irrotational and slow compared to the speed of sound, the Navier–Stokes equations reduce to Laplace’s equation.

In many applications so-called panel methods (or boundary element methods) have become the standard approach to solve Laplace’s equation. The time it takes for a good implementation on a modern computer system to solve, for example, the flow over an airfoil is on the order of milliseconds. This makes such panel methods well suited as the inner solver in an optimization algorithm. In this context, the goal of the computer program is to find an ideal airfoil geometry given a target function. The suggested algorithm requires two computationally expensive steps: the assembly of the matrices for a given geometry and the solution of the resulting linear system of equations.

The goal of this talk is to evaluate the performance of the Intel Xeon Phi 7120 and the NVIDIA K80 to accelerate this algorithm. For that purpose, we have implemented an optimized version of the algorithm on the CPU and Xeon Phi (based on OpenMP, vectorization, and the Intel MKL library) and on the GPU (based on CUDA and the MAGMA library). Since the Xeon Phi/GPU is more efficient for the assembly step but the CPU is more efficient for the linear solve, this algorithm benefits from a heterogeneous computing environment.

We present timing results for all codes and discuss the similarities and differences between the three implementations. Overall we observe a speedup of approximately 2.5 for adding an Intel Xeon Phi 7120 to a dual socket workstation and a speedup between 3 and 3.5 for adding an NVIDIA K80 to a dual socket workstation.
VSC School Project:  
Evaluation of the Partitioned Global Address Space (PGAS) model for an inviscid Euler solver

Martina Prugger, Lukas Einkemmer, and Alexander Ostermann  
Department of Mathematics, University of Innsbruck

The modeling of fluids usually results in a number of partial differential equations that relate the change of local properties (such as density, velocity, temperature, ...) in time to the corresponding change in space. Among the equations used, the Euler equations (for inviscid flow) and the Navier-Stokes equations (for viscid flow) are probably the most prominent examples and are used in a variety of applications. Mathematically the proper discretization of conservation laws is of importance to obtain physically relevant results that can be used in applications ranging from the analysis of aircrafts to transport phenomena in the sun.

These problems are challenging from a numerical point of view, since care has to be taken to propagate shock waves without diminishing the performance of the scheme. Even though a couple of software packages have been developed that are used in both an industrial as well as in an academic setting, considerable progress is still to be made in the design of numerical integrators and their parallelization to large scale computer systems. Due to the discontinuities present in the solution, linear and monotone schemes are at most first order accurate (Gudonov’s theorem). This has stimulated research in both the construction of limiters as well as in using the (for some equations) known analytical solution of the Riemann problem in a single space dimension in order to approximate the three dimensional solution. However, significant research remains to be conducted with respect to both the construction of viable high order schemes as well as their parallelization.

Message passing with MPI is the classical approach for high performance computing on clusters. However, in recent years, partitioned global address space (PGAS) languages have emerged that offer an incremental approach for the parallelization on cluster and hybrid shared/distributed memory systems. This offers the opportunity for the programmer to optimize the parallel code step-by-step until the desired level of scaling is achieved. In addition, PGAS languages inherently use one-sided communication primitives and are thus conjecture to offer better scalability on exascale systems.

In this work, we consider a two dimensional fluid dynamics code, where we use finite volume methods to discretize space and Godunov’s method as well as an exact Riemann solver to capture shocks. We have implemented our solver in both MPI and Unified Parallel C (UPC). The latter is an extension to the C programming language that implements the PGAS paradigm. We provide comparative scaling studies on different computer systems of the Austrian HPC infrastructure (namely, the LEO3 and LEO3E systems at the University of Innsbruck, and the VSC-2 and VSC-3) and describe the incremental parallelization that has been done for our application.

References

DASH: data structures and algorithms with support for hierarchical locality

Karl Fürlinger, Tobias Fuchs, and Roger Kowalewski

Ludwig-Maximilians-Universität München, Germany

DASH [1] is a data-structure oriented C++ template library under development in the context of SPPEXA, the priority program for software for Exascale computing funded by the German research foundation (DFG). DASH is a realization of the PGAS (partitioned global address space) model using operator overloading to provide global-view PGAS semantics without the need for a custom PGAS (pre-)compiler. The DASH library is implemented on top of our runtime system DART [2], which provides an abstraction layer on top of existing one-sided communication substrates. DART contains functionality to allocate memory in a global address space as well as collective and one-sided communication primitives. To support the development of applications that exploit a hierarchical organization, either on the algorithmic or on the hardware level, DASH features the notion of teams that are arranged in a hierarchy. Based on a team hierarchy, the DASH data structures support locality iterators as a generalization of the conventional local/global distinction found in many PGAS approaches.

The simple example on the left shows a small stand-alone hello world DASH program that allocates a small 1D array of integer keys and stores them over all available nodes. DASH follows the SPMD (single program, multiple data) model and the execution environment is initialized by the `dash::init()` call in line 3. Subsequently, `size` gives the number of participants in the program (called units) and `myid` identifies an individual unit. As an extra benefit of using DASH, rather than a local container such as a Standard Template Library (STL) vector or array, the storage space is not limited by the locally available memory, but is extensible by adding more resources in a distributed memory setting. In the example code the DASH array allocated in line 7 is used to communicate a single integer key from unit 0 to every other unit in the application. The communication is accomplished by overloading the subscript operator ([ ]) of the `dash::array` container and in lines 10–11 unit 0 stores the key at every (distributed) memory location of the array. The default layout for DASH one-dimensional arrays is blocked over all units. In our example this mapping implies that `key[i]` is stored on unit `i` and hence the access in line 16 (`key[myid]`) does not generate a communication event, since every unit reads its own local data item.

A DASH-enabled application can use the data structures and programming mechanisms provided by DASH. An application can be written from scratch using DASH, but we envision that more commonly existing applications will be ported to DASH, one data-structure at a time.

References


VSC School Project:

Optimized sparse matrix-matrix multiplication for multi-core CPUs, GPUs, and MICs

Andreas Morhammer\textsuperscript{a}, Karl Rupp\textsuperscript{a}, Florian Rudolf\textsuperscript{a}, and Josef Weinbub\textsuperscript{b}

\textsuperscript{a}Institute for Microelectronics, TU Wien
\textsuperscript{b}Christian Doppler Laboratory for HPTCAD, Institute for Microelectronics, TU Wien

Sparse matrices are extensively used in areas such as linear algebra, data mining, or graph analytics. One of the fundamental operations is general sparse matrix-matrix multiplication (SpGEMM), where our primary interest is in computing coarse grid operators in algebraic multigrid methods \cite{trottenberg2001multigrid}. While certain applications provide additional information to derive optimized sparse matrix-matrix multiplications, a fast and general SpGEMM is desirable from an abstraction point of view. As a consequence, parallel implementations of SpGEMM are provided by several libraries including the Math Kernel Library (MKL) by INTEL for CPUs and MICs, and CUSP as well as CUSPARSE by NVIDIA for NVIDIA GPUs.

In this work we present optimization results for SpGEMM on shared memory systems equipped with multi-core CPUs, GPUs, or MICs. We build on top of previous work on optimizing SpGEMM for NVIDIA GPUs \cite{gremse2015gpu}, generalize the optimization techniques to other architectures \cite{rupp2016viennclassparse}, and derive fast implementations for hardware from all major vendors: First, an SpGEMM kernel implementation based on second-generation advanced vector extensions (AVX2) intrinsics merging multiple rows concurrently on the latest Haswell Xeon CPU line, and an implementation based on 512-bit wide AVX intrinsics on Xeon Phi (KNC) is discussed. Second, an embedded performance model for estimating the work required by each thread is introduced, resulting in improved load balance across threads. Third, our contribution for a GPU-based SpGEMM is a refinement of the recently proposed row-merging algorithm proposed in Ref. \cite{gremse2015gpu} by reducing the memory footprint and the number of kernel launches. While the original row-merging algorithm has memory overheads of at least the size of the result matrix, the additional memory required by our algorithm depends only on the total number of threads and the maximum number of nonzeros in the right hand side factor matrix. A comparison with MKL, CUSP, and CUSPARSE in Figure 1 demonstrates a 50 percent performance gain over INTEL’s MKL library on a recent Haswell-based Xeon system on average. A two-fold performance gain over CUSP and CUSPARSE is demonstrated on an NVIDIA Tesla K20m. Also, we present the first implementation of an efficient SpGEMM on AMD GPUs based on row-merging.

\begin{figure}
\centering
\includegraphics[width=\textwidth]{fig1}
\caption{Performance comparison of sparse matrix-matrix multiplication routines.}
\end{figure}

References

\begin{itemize}
\end{itemize}

23
A computational scientist’s perspective on current and future hardware architectures

Karl Rupp\textsuperscript{a} and Josef Weinbub\textsuperscript{b}

\textsuperscript{a}Institute for Microelectronics, TU Wien
\textsuperscript{b}Christian Doppler Laboratory for HPTCAD, Institute for Microelectronics, TU Wien

Power constraints prohibit further increases in clock frequency and thus single-threaded performance [1]. As a remedy, hardware vendors equip their processors with multiple cores to further increase the overall computational power provided. However, raw computational power can only be leveraged if data can be accessed and moved quickly. On the other hand, collective operations such as global reductions either within a single compute node or across a compute cluster are typically limited by latency, which cannot be reduced indefinitely due to fundamental physical limits. To successfully design algorithms and implementations for current and future supercomputers it is mandatory to have a solid understanding of these limits. Most importantly, this requires computational scientists to use parallel algorithms with medium- to fine-grained parallelism already on the node-level. Finding and exposing such levels of parallelism is, however, often difficult and subject to ongoing research in many application areas [2].

In this talk we evaluate current and future hardware architectures to aid the design of the forth generation of the Vienna Scientific Cluster (VSC-4). Our focus is on limits on strong and weak scalability, synchronization and data transfer latency, arithmetic intensity, as well as available programming models for typical hardware used in high performance computing. We will present benchmark results to quantify these limits and explain the application areas for which the respective benchmarks are relevant.

Overall, our findings confirm that central processing units (CPUs) are best suited for general purpose workloads and are most attractive for investing in long-term code modernization efforts. Graphics processing units (GPUs) and Intel’s many-integrated-core (MICs) devices provide a narrow –yet attractive– sweet spot for applications that are either bound by the floating point operation rate or memory bandwidth (Fig. 1). However, today’s availability of software in science and engineering which can efficiently make use of such many-core platforms is limited as is the experience and skillset among the developers, warranting only a limited availability of supercomputers powered by GPUs and MICs. To date, other accelerator platforms as well as ARM-based hardware cannot be recommended for VSC-4 because of either specialized use cases or lack of maturity.

Fig. 1: STREAM benchmark results obtained for current Intel hardware. While four to eight threads are sufficient to achieve a large fraction of peak memory bandwidth on CPUs, MICs require at least 64 active threads to achieve high memory bandwidth.

References


FELTOR: a hybrid MPI + OpenMP/GPU code for plasma turbulence simulations

Matthias Wiesenberger, Markus Held, and Alexander Kendl

Institute for Ion Physics and Applied Physics, University of Innsbruck

The simulation of fully developed three-dimensional turbulence in magnetized plasmas requires the efficient usage of computational resources. Our code FELTOR (Full-F ELectromagnetic model in TORoidal geometry; F stands for the probability density function F in phase space) uses a design principle based on template meta-programming (TMP) in C++. In recent years, this technique has found applications in the scientific community under the name container free numerical algorithms. The idea behind the design is to separate numerical code, the “front end“, from the specification, optimization, and implementation of vector and matrix data structures and operations, which we call the “back end“. This is done by formulating numerical algorithms using only a limited set of templated functions that execute elementary operations like vector addition, dot products or matrix vector multiplications. In principle our code can run unchanged with any container class that provides the necessary back end functionality. We have implemented single core CPU, multi core OpenMP and GPU back ends through the use of the CUDA thrust library. Also an MPI back end is available, which in turn makes use of the OpenMP and GPU code to generate MPI + OpenMP/GPU code.

We show results from performance tests on VSC-2, VSC-3 and on our own GPU cluster SERLES. In general we find the use of GPUs superior to CPUs for memory bound problems like ours. We also find that for our cases the NVIDIA gaming cards provide the same or even better performance than the corresponding scientific cards, which are many times more expensive.

The high degree of parallelism in our algorithms stems from the use of discontinuous Galerkin (dG) methods. We have used and adapted these methods for FELTOR in Reference [1,2]. In magnetized plasmas the dynamics parallel and perpendicular to the magnetic field lines follows different length and time-scales. This is usually exploited by aligning the numerical grid with the magnetic field, which however exhibits northogonal and anisotropic meshes near the last closed flux surface (LCFS) as well as singular points. Our three-dimensional computations rely on a flux-coordinate independent (FCI) approach, which overcomes these difficulties, while still allowing minimal parallel resolutions [3]. We demonstrate this with full-f gyrofluid turbulence simulations in global axisymmetric toroidal X-point geometry.

References
Computational astrophysics: planets, stars and galaxies

Christian Rab\textsuperscript{a}, David Bancelin\textsuperscript{a}, Colin Johnstone\textsuperscript{a}, Thomas I. Maindl\textsuperscript{a}, Christian Sch"afer\textsuperscript{b}, Patrick Steyerleithner\textsuperscript{a}, and Eduard I. Vorobyov\textsuperscript{a}

\textsuperscript{a}Department of Astrophysics, University of Vienna
\textsuperscript{b}Institut für Astronomie und Astrophysik, Universität Tübingen, Germany

In astrophysics we deal with a large variety of different objects (e.g. particles, planets, stars and galaxies) and physical phenomena (e.g. N-body interactions, fluid dynamics, radiative transfer, chemistry and solid-state physics). At our institute we use computational methods to study the habitability of planets, star-formation and the evolution of galaxies. We will present an overview of the various projects and discuss the different codes and methods used.

**Planets and Habitability:** We use N-body simulations to investigate the transport of water from e.g. the asteroid belt to the inner region of planetary systems, with a focus on binary systems. To simulate the interactions of these "small" bodies we apply a smoothed-particle hydrodynamics (SPH) code to investigate the impact of collisions on the water transport. This OpenMP based SPH code was recently ported to CUDA to make use of the computational power of graphic cards. Also the star planet interaction is relevant for habitability. With a 3D hydrodynamics code we model the wind interaction in binary systems to study the consequences for habitability (e.g. impact of shock waves).

**Star Formation and Disks:** Protoplanetary disks, the birthplaces of planets, are formed around low mass stars. To study the formation and long term evolution of disks (starting from the collapse of the parent cloud), we use a hydrodynamics code particularly developed for disk modelling. With a so called radiation thermo-chemical disk code we investigate the detailed thermal and chemical structure of disks. This code is especially used for fitting observations (large number of models). Currently both codes use OpenMP for parallelization. However, current developments focus on combining these two methods (e.g. include chemistry in the hydro code) which will significantly increase the computational needs.

**Galaxies:** To simulate the interaction of dwarf galaxies with the surrounding material (e.g. galaxy clusters) we use an adapted version of the massively parallel adaptive mesh refinement (AMR) code FLASH. With this code it is e.g. possible to calculate the mass loss of dwarf galaxy caused by Ram pressure stripping. In this context also a novel chemodynamical code was developed at our institute. This code uses a "stellar hydrodynamics" approach to model the stellar component. In future simulations this method will be used in the FLASH code to make use of the well parallelized AMR technique.

*The computational results presented have been achieved in part using the Vienna Scientific Cluster (VSC).*

![Particle collision](image1.png)
![Disk formation](image2.png)
![Galaxy stripping](image3.png)

Fig. 1: Examples of various simulation results. Note the different time and spatial scales.
VSC School Project:

Introduction to the Vienna Protein Simulator: performance and applications

Francesca Nerattini, Luca Tubiana, and Ivan Coluzza

Computational Physics, University of Vienna

The Vienna Protein Simulator (ViPS) is a Monte Carlo simulation package for protein folding and protein design which uses a novel coarse-grained protein model: the Caterpillar [1]. Protein folding and design are major biophysical problems, the solution of which would lead to important applications in several fields, for example biomedical research, drug discovery, molecular biology, to mention only a few.

The ViPS package will provide a stable, fast and reliable computational tool to research groups who study the challenging problems of protein folding and design. The ViPS will allow researchers to substitute in house code with an optimized and well tested simulation package.

The ViPS package uses a Virtual Move Parallel Tempering scheme (VMPT)[2] to enhance sampling of the free energy landscape of proteins. VMPT is a highly efficient scheme that makes optimal use of parallel machines, running copies of the same simulations at different physical parameters (e.g. temperature or pressure) and collecting information from rejected Monte Carlo moves as well as from accepted ones.

Overhauling, analyzing and tuning of scientific software are the key objectives in all VSC School Projects, hence current efforts within the ViPS project may be regarded a prime example of modern code optimization strategies [3]. Of particular importance are modularity, flexibility, scalability and preparedness for likely occurring future extensions, hence a fundamental need is the proper documentation of individual steps within the software innovation cycle.

Here we present the structure of the ViPS package, its use and the improvements made within the original code in terms of reliability, documentation system and modularization. We will show the results of our performance profiling of ViPS as well as the increasing in the sampling power of the code as a function of the number of processors used (see Fig. 1).

![Fig. 1: Sampling efficiency as a function of the number of cores with and without VMPT scheme. The sampling efficiency has been evaluated from the statistics of a replica with a temperature common between all the simulations. Insets: Free-energy landscapes computed with the VMPT algorithm using 2 and 16 cores. Making use of the VMPT scheme clearly improves the sampling efficiency which increases with the number of cores involved.](image)

References

Molecular dynamics simulation of CD8-MHC interaction

Claudia Stocsits, Reiner Ribarics, Rudolf Karch, and Wolfgang Schreiner

Section of Biosimulation and Bioinformatics, Center for Medical Statistics, Informatics and Intelligent Systems (CeMSIIS), Medical University of Vienna

The immune system of vertebrates can be classified into innate and adaptive forms of immune response. Innate immunity builds a first immediate line of protection against pathogens. All multi-cellular organisms possess multiple components of the innate immune system, e.g. phagocytic cells, antimicrobial peptides and the alternative pathway of complement activation.

In contrast to the generic defense mechanisms of the innate immune system, the adaptive immune system specifically adapts in response to the recognition of a pathogen and the efficiency and defensive capabilities of the immune response increase with each exposure to a particular pathogen. A key role in the adaptive immune response plays the interaction between the T-cell receptor (TCR) and the major histocompatibility complex (MHC). MHC proteins are surface-bound proteins that present short peptide fragments (p) in a cleft formed by two α-helices on the surface of infected cells or tumor cells. TCR recognizes specifically the peptide presenting α-helices and the peptide. The recognition of the MHC bound peptide is assisted by the binding of the co-receptor Cluster of Differentiation 8 (CD8). CD8 is a T-cell surface glycoprotein which forms a dimer that recognizes the α3-domain of the MHC molecule. It is known from the literature that TCR-pMHC interaction enables specific binding, whereas pMHC-CD8 interaction enhances the sensitivity of binding by affecting the TCR-pMHC association rate and stabilizing the interaction of TCR-pMHC [1].

Molecular dynamics (MD) [2] is a valuable tool for simulations of physical movements of atoms and molecules by solving Newton’s equations of motion. Previous work by our group analysed the geometric dynamics of MHC α-helices in different major histocompatibility complexes [3]. The aim of the present study is to investigate functional molecular elements and dynamic molecular processes in a set of three closely related MHC molecules, i.e., human leucocyte antigen subtype B: HLA-B*44:02, HLA-B*44:03, and HLA-B*44:05. The structures of TCR-pMHC (HLA-B*44:05) and pMHC-CD8 are resolved and available at the RCSB Protein Data Bank (www.pdb.org). In addition, homology modelling was used to obtain 3D structures of the molecules (HLA-B*44:03, HLA-B*44:02). We use molecular dynamics (MD) to simulate the binding of CD8 to the above-mentioned MHC molecules in absence and in presence of TCR molecule at an atomic level of detail to characterize the orientation and movements of MHC domains.

References


The aim is to calculate and reliably predict properties of solid matter free from any empirical parameter purely based on the quantum physical interaction of electrons and nuclei as comprised in Schrödinger’s equation. Hence, such methods are called “ab-initio” or “first-principles” methods. The vast majority of computer codes and applications in this field are based on the concepts of density functional theory (DFT) for which its orbital based formulation (Kohn-Sham equations) is utilized [1]. By that, Schrödinger-like equations have to be solved selfconsistently for the orbitals which only serve as auxiliary quantities for constructing the ground state charge density and minimizing the ground state total energy.

As demonstrated by the ever increasing number of publications DFT approaches are very widely used for a large range of applications in pure science as well as in more applied fields. For numerical computations suitable numerical/analytical orbitals or basis functions have to be chosen, which characterize a variety of numerical methods. In a solid with its rather compact distribution of atoms space may be divided into regions between the nuclei (“out”) and close to and around the nuclei (“in”). In region “out” the electron density and potential is rather smooth whereas it varies rapidly in region “in”. This leads to a variety of concepts and ansatzes for designing suitable basis functions, which are then needed for building the Hamiltonian matrix elements. The most accurate numerical methods are able to solve the Kohn-Sham equations without any further modeling restrictions such as assumptions about the geometrical shape of the electron density and potential.

In most of the methods periodic boundary conditions are chosen assuming a crystalline arrangement of the atoms on a lattice. A natural choice for these conditions are plane waves which contain the information about the periodicity of the lattice via the reciprocal lattice. Assuming that plane waves can be used for the basis functions as well as density and potential in the whole space (regions “out” and “in”) then fast Fourier transformations can be applied to build up the Hamiltonian, and the main work consists in solving large eigenvalue problems. Dimensions of the Hamiltonian may be of the order $10^4$ to $10^5$ depending on the number of atoms in the unit cell. However, the strong variation of the potential in region “in” and the consequently bad convergence of its Fourier expansion enforces the construction of so-called pseudo-potentials, which model the potential of the nucleus screened by the electronic core (innermost) states. The construction of good pseudo-potentials is an art for itself, which is more or less established nowadays. All so-called plane wave methods rely on them. One example for them is VASP which is widely used on the VSC systems.

Another concept for designing basis functions consists in using natural expansions, namely plane waves in region “out” and atomic-like functions in region “in”. At a chosen spherical boundary these two sets of functions are then matched continuously. This is the concept of the Augmented Plane Wave (APW) method which is able to deal with general shapes of potentials and charge densities, and it handles full potentials by calculating selfconsistently also the electronic core states. Although the Hamiltonian matrix sizes for APW are typically smaller compared to plane wave methods APW is more time consuming due to the construction of the Hamiltonian in region “in”. On the other hand only the charge and position of the nuclei are needed as basic input. The APW concept is used by WIEN2k also running heavily on the VSC systems.

Other concepts of constructing basis functions consist in using functions localized at the atomic positions (well suited for region “in”) and treating their behaviour and overlap in space “out” in some suitable way. A method typical for such an ansatz is the so-called LMTO method based on atomic-like “muffin-tin” orbitals. Localized Gaussian functions suitably symmetrized for the periodic boundary conditions are also used, as utilized in the Gaussian package. The advantage of Gaussian orbitals is that they can be used also for
molecules with atomic-like boundary conditions (the basis functions decaying properly fast with distance) whereas codes with periodic boundary conditions can only rely on suitably large super-cells for describing isolated objects, such as a molecule or an impurity in a perfect crystalline lattice.

For the description of such a symmetry breaking by a localized perturbation (which destructs the periodic boundary conditions) Green’s functions techniques would be conceptually well suited. However, such concepts are used rather rarely nowadays, mainly so because so far they are less accurate concerning the calculations for general crystal structures, the energetics and the optimization of geometry. The standard methods utilizing periodic boundary conditions are well able to reproduce in many cases experimental findings such as for example the positions of atoms on a surface. The power and reliability of predicting geometrical parameters is an important topic as utilized for example in combined experimental-theoretical SFB projects for which large computer resources are consumed on the VSC clusters.

The key quantity of DFT is the total energy of the ground state and the requirement on the codes and methods is to deliver reliable numbers for the energetical properties of solid materials. Fortunately, measurable energetical quantities always consist of differences of total energies by which errors inherent in the approximations of the many-body functional may cancel out to some extent. Nevertheless, for a variety of (correlated) systems these approximations are significantly bad and a lot of present and future efforts are and will be put on the improvement of the many-body approximations, ranging from semi-empirical adjustments to efforts towards constructions of “exact” wave functions [2]. This topic is also important for a proper description of the electronic properties of a crystalline solid such as the gap between valence and conduction states. Presently, many applications on the VSC are using improved many-body functionals requesting large computer resources.

Major future applications for materials properties will deal with larger systems (larger number of atoms per cells), complex properties (many-body interactions, electron-phonon coupling, magnetic ordering) and more complex derived properties (vibrational properties, thermal transport). A further subject for seminal applications requiring large computer resources is the scanning of large configuration spaces for modeling structural, thermodynamical and other temperature dependent properties of alloys and magnetic phases. This requires a large set of DFT calculations in combination with Monte Carlo calculations which involve multi-body potentials [3].

References

VSC School Project:
Evaluating the scaling behavior of WIEN2k

Thomas Ruh and Peter Blaha
Institute of Materials Chemistry, TU Wien

Introduction: The program package WIEN2k [1] utilizes Density Functional Theory (DFT) to describe materials and to calculate their properties employing quantum mechanical simulations. During these simulations the Schrödinger equation of a given system is solved numerically. The main part of these calculations is solving a hermitian (or symmetric) generalized eigenvalue problem involving relatively large matrices (up to dimensions of several 100,000 depending on the system).

Nowadays “simple” materials (with about 50–100 atoms in the unit cell and structures with inversion symmetry) can easily be simulated on a modern PC without too much effort. Bigger systems or more complicated structures, however, are much more expensive – both in computation time and in memory consumption:

For instance calculations of surfaces (needed to explain experimentally found reconstructions [2]) need supercells with hundreds of atoms. NMR- and Knight-shifts of metals are a good example for properties that require computationally exhaustive calculations, since they require the solution of the Schrödinger equation at millions of k-points [3].

Aim of the project: To handle this growing complexity of materials and properties (and the resulting large matrices that need to be computed) it is vital to utilize HPC systems like the VSC, in order to yield results within reasonable times and to allow simulation of more complex or bigger systems.

The core programs of WIEN2k are already parallelized and allow the efficient use of large numbers of cores – WIEN2k scales reasonably well up to 128 to 512 cores depending on the problem size: Figure 1 shows a comparison between the scaling results of a medium sized matrix both on VSC-2 and VSC-3 using the code as it is.

However, parallelization beyond 512 cores fails due to bottlenecks, for example during the matrix diagonalization (which is done fully or iteratively using LAPACK and ScaLAPACK routines) or necessary disc I/O. These bottlenecks have to be resolved to improve the performance. Moreover, the adapted code must be reevaluated to possibly identify further, so far unknown, bottlenecks.

Future work: In order to test WIEN2k on different hardware, extensive benchmarks both on VSC-2 and VSC-3 will be performed, simulating materials of varying complexity (leading to real and complex matrices differing in size). In addition alternatives to ScaLAPACK (like ELPA or PLAPACK) will be tested, as a first approach to remove the matrix diagonalization bottleneck.

References
Optimizing the mechanical properties of materials via grain boundary segregation

Lorenz Romaner, Daniel Scheiber, Vsevolod Razumovskiy, and Jürgen Spitaler

Materials Center Leoben Forschung GmbH

Segregation of solute elements to grain boundaries (GBs) is a key factor for the production and the performance of many technologically relevant materials. It influences fundamental properties such as formability, crack propagation, grain growth, precipitation, diffusivity or electric conductivity. Hence, segregation can be used to tailor material behavior and to achieve higher performance. The challenge in this connection is the knowledge and control of the segregation state. Experimentally, only advanced techniques are able to resolve the atomistic structure of GB segregation and a precise characterization of only one (or a few) GBs might not give a representative picture. For these reasons, atomistic simulations of GBs are of high value since they allow conveniently looking into the details of segregation and scanning over a wide range of GB geometries.

In this talk we will present ab-initio calculations of GB segregation for a variety of materials including the transition metals, aluminum and zinc oxide. Computations were carried out with the vienna ab initio simulation package (VASP) on supercomputers including the VSC. The procedure and the demands of these simulations will be described. The first task is the search of the geometric ground state structure of the clean GBs. Due to the high structural complexity of GBs, this search is an involved task and is tackled via a gamma surface approach. On the basis of the ground state structures the calculation of the segregation profile can be carried out. As shown in Figure 1, several different types of segregation sites exist in GBs and their segregation energy can vary strongly. With the treatment of many GBs and GB sites is possible to extract the overall segregation behavior. Furthermore, there exist characteristic trends related to the solute position in the periodic table which we will highlight. Finally, we discuss the implications of segregation for the mechanical properties of the bcc transition metals. We will focus on the ductile-to-brittle transition as well as the fracture mode which can be intergranular (along GBs) or transgranular (through the grain). Our calculations reveal which solutes should be used to strengthen the grain boundaries of these materials. In this way they give hints how intergranular fracture can be reduced and the low-temperature ductility increased.

Fig. 1: Illustration of the atomic structure of 6 different grain boundaries in Tungsten. The Rhenium segregation energy is indicated by the color-scale on the right where darker corresponds to stronger segregation.
Computational material studies on hybrid interfaces: from structure search to property prediction

Oliver T. Hofmann and Egbert Zojer

Institute of Solid State Physics, Graz University of Technology, NAWI Graz

Hybrid electronic devices based on combinations of inorganic and organic materials become increasingly important for high-tech products. Displays are already commonly found in mobile phones and some TVs. Smart clothing products are frequently introduced at fairs. Other, more exotic applications are still at the conceptual state. The common bottleneck for all these applications are the interfaces between the inorganic and organic materials, over which charge or energy have to be transported.

The detailed properties of these interfaces depend sensitively on their structure. Since in devices these structures are buried, i.e., covered by other materials, typically experimental structure determination methods cannot be applied. Therefore, most studies must rely on computer simulation to determine the interface arrangement. So far, most theoretical studies that consider these interfaces from an atomistic perspective have mainly focused on idealized, perfectly ordered interfaces. However, in reality, even if every effort is made to keep the interface well-defined, temperature and entropy will cause the formation of defects in the organic material. These defects can take several guises, from vacancies (i.e., the absence of individual molecules) to patches with a completely different orientation of the molecules. Such defects can and do significantly affect how charge and energy is transported across the interfaces.

In the present contribution, it will be shown how high-performance, parallel computing can be exploited to efficiently search through the multitude of possible structures and defect complexes obtaining both the energetically most favorable geometry and its electronic properties. The largest challenge of this endeavor is the vast configurational space spanned by the various adsorbate morphologies, which we tackle using a divide-and-conquer-approach: First, possible adsorption structures for single, isolated molecules on the surfaces are determined. Then more complex, densely packed layers are modelled, starting from a regular arrangement of the various individual adsorption geometries. The various permutations for such arrangements serve as guess for basins of the potential energy surface, which can then be sampled using a basin hopping algorithm. Unambiguously assigning the different basins allows for a particularly efficient screening that avoids recalculating known structures while allowing to cross parts of the potential energy surface that have already been visited. The applicability and efficiency of this approach will be demonstrated on two conceptually different systems: A well-ordered layer of electron donors and acceptors with a variable mixing ratio [1], and a mostly disordered material that consists of small organic molecules (shown in Figure 1).

Parts of the calculations have been performed using the resources of the VSC-3.

Fig. 1: Examples of defects in organic monolayers: (a) misorientation, (b) broken bonds, (c) dislocations, (d) voids.

References

Core-level calculations with hundreds of atoms using GPAW

Toma Susi

Disorder in Materials, Physics of Nanostructured Materials, University of Vienna

X-ray photoelectron spectroscopy (XPS) combined with first principles modeling is a powerful tool for determining the chemical composition of novel materials. However, many interesting systems contain low concentrations of defects or impurities, requiring very large unit cells for their realistic modeling. At the same time, the need to treat core levels explicitly typically precludes large system sizes. We have found that the projector-augmented wave (PAW) formalism, specifically using the highly scalable GPAW code, allows relatively accurate core level calculations for hundreds of atoms.

As a test case, we calculated the carbon 1s core level binding energy of pristine graphene using two methods based on density functional theory total energy differences: a delta Kohn-Sham calculation with a frozen core-hole (KS) [1], and a novel all-electron extension of the delta self-consistent field (SCF) method [2]. Although the binding energy depends on the chosen exchange correlation functional, using the Perdew-Burke-Ernzerhof functional we find a value remarkably close to what has been measured for graphite. Notably, full convergence of the energies requires well over 100 atom unit cells to minimize spurious interactions between images of the screened core hole.

As a more demanding application, we have analyzed XPS measurements of free-standing powders of C60 and C59N fullerenes using core-level calculations [3]. To model oxygen impurities, including water absorbed in the pores of crystallites, we had to simulate a system of over 400 atoms to establish a vacuum reference for H2O in the same energy scale. We were also able to quantify the experimentally inaccessible effect of C59N dimerization on its N 1s core-level shift (−0.4 eV), with implications for interpreting measurements of other nitrogen-doped systems.

Funding from the Austrian Science Fund (AM0149721 and P28322-N36) and computational resources from the Vienna Scientific Cluster are gratefully acknowledged.

References

VSC School Project:

Diagrammatic quantum Monte Carlo for strongly correlated electron systems

Patrik Gunacker\(^a\), Markus Wallerberger\(^a\), Andreas Hausoel\(^b\), Giorgio Sangiovanni\(^b\),
and Karsten Held\(^a\)

\(^a\)Institute of Solid State Physics, TU Wien

\(^b\)Institute for Theoretical Physics and Astrophysics, University of Würzburg, Germany

Within the international theoretical solid state community, Vienna is famous for its well-established software packages Wien2k, Wien2Wannier and VASP. These software packages rely on density functional theory (DFT) and accurately simulate a wide range of materials on an effective one-particle level. Only recently the w2dynamics software package [1] extended the former list by implementing the dynamical mean field theory (DMFT) [2] in order to allow for a true many-body treatment of strongly correlated electron materials and model Hamiltonians. Already a large share of the simulation time on the Vienna Scientific Clusters results from the w2dynamics code, making optimization strategies crucial. Both ab-initio calculations of materials with partially filled d- and f-orbitals (DFT+DMFT) [3] and model calculations of the Hubbard model and the Anderson impurity model provide a deeper understanding of electron correlation effects. As a result new materials and physics such as superconductivity in cuprates, topological insulators, and correlation driven transistors can be addressed.

The computational heart of the DMFT loop is the dynamic (i.e. time-dependent) quantum impurity problem. The mathematical description of these quantum impurities heavily relies on the Green’s function formalism, while the physical description is usually given in terms of Feynman diagrams. A highly efficient way of numerically solving quantum impurities makes use of the Monte Carlo method. On the one hand, the Monte Carlo sampling is formalized to sampling Feynman diagrams of arbitrary order and on the other hand, the measurement procedure extracts thermal impurity Green’s functions (i.e. the one/two-particle Green’s function, which depends on one/three frequency arguments and two/four spin-orbital indices). When embedding the impurity Green’s function into DMFT, one is able to extract the local self-energy of the system, which is then a physical measure of electron correlations.

In this talk we give an introduction to the quantum Monte Carlo impurity solver from a computational viewpoint. We revisit some basic concepts of Monte Carlo with respect to parallelization. We then focus on optimization strategies in the existing w2dynamics software package. Specifically, we first discuss improved scalings of the algorithm resulting from the implementation of delayed multidimensional FFT algorithms. Further, we exploit different symmetries (time-reversal, SU(2), crossing, ...) of the quantum impurity problem to implement sparse-matrix-like storage and speed- and memory gains. We quantify the optimizations with respect to model calculations, benchmarking against previous releases.

References


High performance computing at IST Austria:
modelling the human hippocampus

Alois Schlögl and Stephan Stadlbauer

IST Austria, Klosterneuburg

Current state: The scientific computing cluster at IST Austria consists currently of about 70 compute nodes mostly from Supermicro. This includes 1 GPGPU node, three nodes with 512 GB RAM, as well as a few NFS storage servers with a capacity of about 150 TB. All nodes are connected through QDR 40 GB Infiniband. Just recently, all compute nodes have been upgraded to Debian 8. For job scheduling, GridEngine is currently used.

In the last 12 month, the cluster has been used by about 50 users from various disciplines, including biology, neuroscience, physics, and computer sciences. The requirements vary greatly. There are jobs which require more than 50 GB RAM per CPU core, OpenMP jobs (e.g. gromacs), as well as OpenMPI jobs, and hybrid (MPI/OpenMP) jobs. Currently, more than 35 different software packages are installed, and managed through the "Lmod: Environmental Modules System". Because many packages have more than one version installed, over 70 different packages are currently available. In addition, numerous scientific software packages from Debian 8 are available.

Future plans: A cooperation with VSC has been started, and a number of selected users have test accounts since autumn 2015. The selection has been done based on the criteria of high CPU usage with an efficient RAM/CPU core ratio, and to have software requirements that are already fulfilled on VSC-3. Users that have more demanding requirements like very high storage volume or require more than 16 GB RAM per CPU core, have a need of proprietary licenses (e.g. Matlab, Mathematica), or have a very difficult software setup, are currently not included in that program. A status update of the user experience will be presented.

Modelling the human hippocampus: The hippocampus is a brain structure that seems to be important for learning. For estimating the storage capacity, a model based on pattern storage and pattern retrieval has been proposed. Based on physiological evidence, the structure of neurons seems to be quite regular and there are estimates that the hippocampus contains about 330 000 neuronal cells. These cells have on average about 20 000 dentrites, connected to other cells in the hippocampus. A number of parameters as well as the question whether a random structure or a more regular structure has a higher storage capacity is addressed. A model has been implemented on a realistic scale containing all 330 000 neurons and its possible connections between all neurons.
Responding to HPC needs in Scientific Network South Tyrol:
HPC for computational linguistic purposes

Samuel Senoner

Information and Communication Technologies, EURAC Research, Italy

An initiative of the Provincial Government of South Tyrol, EURAC was founded in Bolzano in 1992 as a private association. The full name is "European Academy for Applied Research and Education Bolzano". EURAC was established in South Tyrol, a mountainous, multilingual region on the border between Italy and Austria, where Italian-, German- and Ladin-speaking communities live and work in a spirit of cooperation. The needs of this region, along with the knowledge it generates, form the basis of EURAC's studies, and its findings are shared at international level. Researchers come to EURAC from all over the world.

In its first year, twelve staff members worked in three areas of research: Language and Law, Alpine Environment, and Minorities and Autonomies. One of the first significant project assignments for the young, rapidly growing research institute was a feasibility study in 1993 to start a new university. This led to the foundation of the Free University of Bolzano in 1997.

The two organizations were strongly connected by their mission, and they created a virtual organization called Scientific Network South Tyrol. Moreover, an important link is also the collaboration in the IT sector, because both organizations IT divisions share the same head of department, Dietmar Laner. In more recent years, new research fields have been added, such as climate change, genetic medicine, applied remote sensing, renewable energy, agricultural and agro-environmental sciences, industrial mechanical as well as energy engineering. All these new areas demand computational resources, which in the past were mainly provided by small systems, local clients and workstations, which, in many cases, required high computational run times. In particular, the increasing role of computational approaches in processing large amount of linguistic data calls for more powerful computational resources and larger storage space.

This contribution will provide an overview of the research areas of the Scientific Network South Tyrol and of the first experiences in using the Vienna Scientific Cluster resources. At the same time, it will discuss high performance computing for computational linguistics purposes.

The Langage Technology group of the Institute for Specialised Communication and Multilingualism is currently involved in a project called OPATCH (Open Access to and Analysis of Text from Cultural Heritage) where an impressive set of 1.8M pages of south tyrolean newspapers from the 19th and 20th century have been processed by optical character recognition (OCR). One key aspect of the project is to annotate the documents with linguistic and semantic information. However, given the derogated quality of the printed paper and the notoriously peculiar fraktur font, the digital output is pervaded with recognition errors and must first go through intensive correction before the annotation can take place. Correcting and annotating 1.8M pages with roughly 1k tokens per page requires high performance computing and the Vienna Scientific Cluster has served the OPATCH project well.

The OPATCH pipeline for correcting and annotating includes scripts in Python and Perl using large dictionaries (around 10M entries) and fuzzy matching through SimString (www.chokkan.org/software/simstring) based on another 10M entries from a database. Part-of-speech annotation (i.e., a grammatical category such as noun and verb) and lemma annotation also make use of a large language model and a Linux-based program called Treetagger (www.cis.uni-muenchen.de/ schmid/tools/TreeTagger). Named entity (i.e., typically names, locations and organizations) annotation is carried out using Java and the Stanford Named Entity Tagger (nlp.stanford.edu/software/CRF-NER.shtml). Finally, the output must be conformed to the latest standard for OCR data including annotations. Needless to say, such a complex procedure on 1.8M documents necessitate a substantial amount of intermediate files and memory management. All in all, the pipeline took around one month to fully complete using on average 1–2 nodes from the cluster.
Evolving computing for particle physics in Austria

Dietrich Liko

Institute of High Energy Physics, Austrian Academy of Sciences

Particle physics has been a driving force to establish a worldwide network of interconnected computing centers around the world. Using the distributed computing resources it was possible to provide the user community a framework for rapid physics analysis [1]. In that context the Institute for High Energy Physics of the Austrian Academy of Sciences is hosting a Tier-2 center [2]. It provides on the one hand the Austrian physics community access to the grid resources and on the other hand it contributes to the overall effort.

New industry trends ranging from cloud computing to big data have not passed unnoticed in this community as well. In particular in the US some effort is going on to study the possibility to deploy particle physics computing on public clouds as AWS (Amazon Web Services). At other places the use of private cloud facilities for scientific computing is seen more relevant. Cloud techniques can provide a flexible setup to share resources with other communities.

An important effort in this area has been the establishment of a remote Tier-0 computing center in Hungary, based on cloud techniques. Using such techniques can lead also to a significant simplification of the workload management of the virtual organizations. Of particular interest is also the choice of the virtualization environment, as for example Docker or KVM.

In the area of storage there are now tools that can be used to complement or even substitute solutions specific to particle physics. Some of these solutions, as HADOOP and CEPH, are already used in sites around the world. But it should also not be forgotten that CERN choose to base its 90 PB disk pool on a home grown solution (EOS). For Austria the evaluation of these technologies is in an early stage.

The picture, that is evolving, is that grid computing technologies will continue to play an important role for the interoperation of particle physics computing centers. Cloud computing on the other hand covers complementary aspects as the deployment and interoperability with other local communities.

Apart from the technological aspects the evolvement of the computing environment for the Austrian particle physics community has also to aim to increase the exchange with partners in Austria. Common interests have to be identified and technology choices have to take into account a bigger picture. This presentation should also facilitate the discussion with other communities to explore common ground.

References

Sentinel big data storage design and benchmarking results

Michael Aspetsberger, Aleksandar Djurdjic, Wolfgang Ebner, Christian Federspiel, Andreas Gerstmayr, and Martin Strigl

Catalysts GmbH, Linz

Representing the latest generation of earth observation satellites, the Sentinel series of satellites will generate a constant influx of earth observation data. When in full operational capacity, 6 satellites will generate 2–3 Terabytes per day each, non-stop, 7 days a week, for a period of at least 7 years. The Earth Observation Data Center (EODC) has set out to acquire, archive, and process all these Petabytes of data.

As a first step, a 5–10 Petabyte archive will be procured and setup. This will be extended incrementally to account for the increasing need while benefiting from the anticipated decrease of cost per byte in the coming years. The archive will be attached to the VSC-3 and future VSC-4 to utilize the massive processing resources for re-processing campaigns. This imposes particular challenges on the interconnection to fulfill the specified target requirements.

We will present the experience gained with the setup of a big data archive for satellite data. We will outline the requirements and design considerations and show the first results gathered when benchmarking a 1 Petabyte partition. The benchmarks cover different storage solutions and usage scenarios. Based on this, we will give suggestions on how to handle data access optimally in end user applications to fully utilize the hardware capacities.
Tutorial:
Parallel file I/O: using and profiling HPC storage

Michael Alexander
VSC Team

While many applications focus on parallelizing computation, the rapidly growing class of throughput-oriented workloads mandates and adds emphasis on parallel I/O. This talk starts with a brief overview of the storage infrastructure on the VSC-3 and VSC-2 clusters, properties of parallel spinning-disk based storage, and typical HPC I/O stacks. Using three application workload types the access methods: POSIX file I/O, HDF5-PnetCDF, MPI/IO, and entity-attribute-value persistence are contrasted. Emphasis is put on the need to understand the runtime behavior with respect to I/O. While difficult in case of standard codes, it may readily show speedup opportunities in self-developed applications.

Characterization and optimization follows, presenting multiple approaches to profiling I/O bound application codes using Darshan, VTune, IPM, and other I/O profiling tools, with and without instrumentation. The notion that there is no one size fits all approach to spotting and improving I/O bottlenecks is stressed. Given sample optimization targets, improvement techniques are highlighted including process-to-file patterns down to block size heuristics.

Furthermore, the talk provides an online view on the emergent behavior of parallel shared environment storage, giving notions of responsible resource usage and having to account for inherent time variances due to changing load patterns. Lastly, an outlook on the coming VSC-3 storage extension and a further smaller extension to the VSC-2 parallel file system along with tool and training resources is given.
Keynote Talk:

Challenges in geo-temporal control of multi-clouds

Ivona Brandic

Institute for Software Technology and Interactive Systems, TU Wien

Introduction: Cloud Computing is a disruptive technology that is gaining momentum globally. In Clouds resources are pooled and offered on-demand with ubiquitous network access to rapidly configurable and elastic IT capabilities. Resources like Virtual Machines (VMs) and Physical Machines (PMs) are managed and coordinated by a Cloud Operating System (Cloud OS). Owing to technological and infrastructure advances over the last years Cloud OSs evolved into complex decision making entities for efficient admission control (e.g., by accepting or denying a service request) and low-level resource management (e.g., starting or stopping VMs).

The fundamental problem in this approach is the lack of appropriate tools, methods and frameworks to efficiently manage and allocate resources considering the balance between the offered services and user expectations. Thus, the general issue by maintaining Quality of Service (QoS) is reduction of cost for the Cloud providers while at the same time maintaining user expectations in terms of quality of delivered services. Nowadays provider costs are boiled down to the energy costs for the operation of data centers.

Multi-Cloud is a novel approach where a single application is deployed on multiple Clouds, i.e., on geographically distributed data centers. Examples for such applications are ultra-scale applications for the management of smart cities, or typically big data applications. Thus, there is additional challenge for handling the additional complexity of coordinating the services across multiple external infrastructure providers, e.g., planning, initiating, monitoring, and enactment of services. Such distributed data center infrastructure changes the rules of cloud control, as energy costs depend on current regional electricity prices and temperatures. Currently, existing cloud controllers are focused on VM consolidation methods suitable only for a single data center or consider migration just in case of workload peaks. Thus, new management approaches for distributed data centers are necessary in terms of resource allocation and pricing policies.

In this talk we discuss challenges when deploying ultra-scale applications on multi-clouds.

Geo-Temporal Inputs: Currently, computation is shaping as a distributed utility whereby costs for computation depend on temporal factors like distributed power generation, microgrids and deregulated electricity markets. The latter have lead to demand of real-time electricity pricing options where prices change hourly or even every minute. Moreover, due to energy overhead range from 15% to 45% of a data center’s power consumption new solutions for cooling of data centers based on outside air economizer technology result in cooling efficiency depending on local weather conditions. In the first part of this talk we discuss the techniques necessary to distribute computation on demand on virtualized geo-distributed data centers considering geo-temporal inputs like time series of electricity prices, outside temperature and similar [1,2].

Pricing in Multi Clouds: The use of virtualization enables on demand resource provisioning including CPU cores, memory, storage, and network bandwidth. Thus, resources are served to customers under pay per use policy. Usage policies are defined through Service Level Agreements - contracts between providers and consumers including type and quantity of resources. While resource quantity is well defined (e.g., through VM flavours) the QoS is usually limited and only restricted to VM availability. However, VM availability does not say anything about the availability of underlying resources like CPU, nor the impact on the performance to the customers applications [3,4]. Thus, in the second part of the talk we discuss a metric that is able to isolate the impact of the resources provisioned to cloud users, hence allowing provider to measure the quality of the provided resources and manage them accordingly.
References


Discrete load balancing in heterogeneous networks with a focus on second-order diffusion

Hoda Akbaria, Petra Berenbrinkc, Robert Elsässerb, and Dominik Kaaserb

a Simon Fraser University, Canada
b University of Salzburg, Austria

In our paper [1] we consider a wide class of discrete diffusion load balancing algorithms. The problem is defined as follows. We are given an interconnection network or a parallel machine and a number of load items, which are arbitrarily distributed among the nodes of the network. The goal is to redistribute the load in iterative discrete steps such that at the end each node has (almost) the same number of items. In diffusion load balancing, nodes are only allowed to balance their load with their direct neighbors.

We present a general framework for randomly rounding the flow generated by continuous diffusion schemes over the edges of a graph in order to obtain corresponding discrete schemes. Our framework can be used to analyze a wide class of diffusion algorithms, supporting heterogeneous networks and second order schemes. Furthermore, we bound the deviation between randomized second order schemes and their continuous counterparts. Finally, we provide a bound for the minimum initial load in a network that is sufficient to prevent the occurrence of negative load during the execution of second order diffusion schemes.

Our theoretical results are complemented with extensive simulations on different graph classes. We show empirically that second order schemes (SOS), which are usually much faster than first order schemes (FOS), will not balance the load completely on a number of topologies within reasonable time. However, the maximum load difference can be further decreased if first order scheme is applied after second order scheme. This is shown in Fig. 1, where simulation results for both FOS and SOS on a two-dimensional torus of size $1000 \times 1000$ are shown. For the simulation, we initialized one designated node with a load of $10^9$ and also simulated switching to FOS after 2500 rounds. Additionally, we investigate the impact of the eigenvectors $V$ of the diffusion matrix on the load vector $x(t)$ in a round $t$ and therefore solve $V \cdot a = x(t)$ for $a$ in every round. Note that the largest coefficient governs the convergence rate in the corresponding round.

Fig. 1: The plot shows a comparison between SOS and FOS of the maximum load minus the average load on a two-dimensional torus of size $1000 \times 1000$. One node was initialized with a load of $10^9$, all other nodes were assigned no load at all.

References

The art of benchmarking MPI libraries

Sascha Hunold, Alexandra Carpen-Amarie, and Jesper Larsson Träff

Research Group for Parallel Computing, Institute of Information Systems, Faculty of Informatics, TU Wien

About: The Message Passing Interface (MPI) is the prevalent programming model used on todays supercomputers, e.g., the VSC-3. Therefore, MPI library developers are looking for the best possible performance (shortest run-time) of individual MPI functions across many different supercomputer architectures. Several MPI benchmark suites have been developed to assess the performance of MPI implementations.

Reproducible Benchmarking: Unfortunately, the outcome of MPI benchmarks is often neither reproducible nor statistically sound. We show which experimental factors have an impact on the run-time of blocking collective MPI operations and how to control them. We also present a new experimental method that allows for reproducible and statistically sound MPI measurements [1]. In particular, we show that MPI run-times measured do often not follow a normal distribution, as many are bimodal distributions (cf. Figure 1).

Synchronizing MPI Processes: Two process synchronization schemes are commonly used in MPI benchmarking: (1) relying on MPI_BARRIER or (2) applying a window-based scheme using a common global time. We show that calling MPI_BARRIER can highly skew processes, see Figure 2. In addition, we propose a novel clock synchronization algorithm that combines two advantages of known algorithms, which are: (1) taking the clock drift into account and (2) using a tree-based synchronization scheme to reduce the synchronization duration [2].

Summary: Our results can help scientists, which work on the VSC-3, to assess the performance of their parallel codes, as our methods guide researchers to identify performance problems in parallel applications.

References


An approach to parallel nonnegative tensor factorization for HPC applications

Markus Flatz and Marián Vajteršic

Department of Computer Sciences, University of Salzburg

The need to process large amounts of data is prevalent in modern society. One important class of data is represented by nonnegative matrices and tensors, which occur in many application areas. The processing and evaluation of such large amounts of data is difficult and time-consuming. Therefore, parallelism is often inevitable to solve such problems in practice.

The goal of Nonnegative Matrix Factorization (NMF) is to represent a large nonnegative matrix in an approximate way as a product of two significantly smaller nonnegative matrices, which are easier to handle and process.

In contrast to other methods such as singular value decomposition (SVD) or principal component analysis (PCA), NMF has the distinguishing property that the factors are guaranteed to be nonnegative, which allows interpreting the factorization as an additive combination of features.

Since matrices can be seen as second-order tensors, NMF can be generalized to Nonnegative Tensor Factorization (NTF) for tensors of any order. The nonnegative input tensor is then approximated by a sum of outer products of nonnegative first-order tensors.

NMF and NTF were successfully used in a variety of application areas, for example in text mining, document classification, clustering, multi-way data analysis, computational biology and neuroscience.

One way to compute a factorization of a nonnegative tensor is to transform the tensor problem into a matrix problem by using matricization [1]. Any NMF algorithm can be used to process such a matricized tensor, including a method based on Newton iteration. In this talk, an approach will be presented to utilize our parallel Newton iteration algorithm for NMF [2] to compute an NTF in parallel for tensors of any order.

The computation was parallelized in two different ways, for distributed systems communicating via message passing using MPI as well as for shared-memory systems using OpenMP. Speedup measurements on the University of Salzburg’s cluster system for relatively large problem sizes with up to 1024 processor cores for the message-passing algorithm and up to 64 processor cores for the shared-memory algorithm will be presented.

References


Surfaces and interfaces of metal oxides are of great importance in many areas of physics, chemistry, and technology; catalysis and electricity generation by fuel cells are just two of them. However, one must never forget that the atomic and electronic structure at surfaces or interfaces very likely differs from a description given by just truncating the bulk, especially if thin films of oxide materials come into play.

Fig. 1: Schematics of the working of a SOFC: $\text{O}_2$ is split and reduced at the cathode, a perovskite oxide surface, and diffuses through the oxide electrolyte (Y-doped Zirconia, YSZ) to the anode. There (oxide/metal interface composite) the $\text{O}^{2-}$ coldly burns $\text{H}_2$, or carbon fuels like alcohols, and generates an electric current flowing in the reverse direction.

Two classes of materials relevant for SOFCs, perovskites and metal/oxide interfaces, have been investigated by employing the Vienna Ab-Initio Simulation Package (VASP), focusing on well characterised systems like the (001) surface of $\text{Sr}_3\text{Ru}_2\text{O}_7$ or the $\text{ZrO}_2$/Pt/Pt$_3\text{Zr}$ interface system. DFT calculations could explain, how a weakly physically adsorbed $\text{CO}$ molecule on the SrO(001) terminated surface of cleaved $\text{Sr}_3\text{Ru}_2\text{O}_7$ is easily transformed into a strongly bound bent $\text{CO}_2$ with the C end bound to the Ru underneath [1]. This strong reaction likely causes the UHV aging effect reported for this and other perovskite oxides. In contrast, an adsorbed $\text{H}_2\text{O}$ molecule dissociates and forms a pair of hydroxide ions $(\text{OH})_{\text{ads}} + \text{O}_{\text{surf}}\text{H}$. Surprisingly, the $(\text{OH})_{\text{ads}}$ stays trapped at Sr-Sr bridge positions, circling the $\text{O}_{\text{surf}}\text{H}$ with a DFT predicted activation energy of $171 \text{ meV}$, matching perfectly the subsequently measured value of $187 \pm 10 \text{ meV}$ [2]. By oxidizing a Pt$_3\text{Zr}(0001)$ surface a closed $\text{ZrO}_2$ film is formed consisting of an O-Zr-O trilayer, reminiscent of a (111) trilayer of the fluorite structure of cubic $\text{ZrO}_2$ [3]. DFT calculations show that the oxide trilayer binds rather weakly to the substrate, rendering the inclusion of van-der-Waals contributions in DFT essential. Experiments and DFT calculations also show that the core level shifts of Zr in the trilayer $\text{ZrO}_2$ films are between those of metallic Zr and thick (bulklike) $\text{ZrO}_2$, casting doubt on previous assignments of such shifts to substoichiometric $\text{ZrO}_x$.

Most of the calculations have been performed using the resources of the VSC.

References
VSC School Project:

Algorithmic challenges in photodynamics simulations

Felix Plasser, Markus Oppel, and Leticia González

Institute for Theoretical Chemistry, University of Vienna

Understanding the interaction of molecules with light is crucial for many areas of contemporary research covering for example the fields of photovoltaics, photobiology, and phototherapy. Experimental investigations are challenging due to the ultrafast nature of the phenomena of interest and simulation provides a vital ingredient to our understanding. Only the smallest model systems can be simulated on standard workstation computers and high-performance computing becomes imperative as soon as more realistic simulations are performed. The dynamical simulation of the processes of interest requires the computation of different terms related to the determination of the wavefunction, the evaluation of the state-to-state transition probabilities, and the computation of the forces acting on the atoms [1, 2]. Unfortunately, the computational characteristics of these steps are entirely disparate as far as the CPU time, the amount of I/O, the memory requirements, and the parallel scalability are concerned. This heterogeneity poses a severe challenge for effective simulations on HPC systems.

In this contribution I will discuss our recent efforts devoted toward the goal of efficient simulations of molecular photodynamics on HPC systems. Practical and algorithmic aspects of the three main computational steps mentioned above are discussed in the context of the multi-reference configuration interaction method. The determination of the wavefunctions is available in a highly scalable implementation, and its viability has already been demonstrated on the VSC in various investigations, see e.g. Ref. [3]. By contrast, the evaluation of transition probabilities, which requires the computation of the overlaps between many-electron wavefunctions amounted to a severe bottleneck until recently. To overcome this problem, a new algorithm was devised that allows storing various recurring intermediates in memory. As shown in Fig. 1, this led to a speed-up of about three orders of magnitude. A downside of the new algorithm are enhanced memory demands, which, however, only come into play for the largest wavefunction expansions. Finally, our strategies for dealing with the last step, the evaluation of the forces, will be outlined.

Fig. 1: Computation times and memory requirements for the old and new wavefunction overlap algorithms in the cases of different wavefunction expansions plotted against the number of terms to be computed.

References
Quantum chemistry and molecular dynamics on supercomputers

Markus Oppel and Leticia González

Institute of Theoretical Chemistry, University of Vienna

Both quantum chemistry and molecular dynamics simulations are at the heart of modern theoretical chemistry. At the same time, both types of molecular simulations are able to challenge even the most advanced supercomputers. This contribution will present benchmark results and experiences from our group on the applications of dynamical and quantum chemical methods on VSC-2 and VSC-3. The efficient implementation of such methods, as well as its combination, allows us to tackle photochemical problems ranging from questions addressing the photostability of DNA [1] to the understanding and design of photochemically active anticancer drugs [2].

In the first part of my talk, a recently developed algorithm called SHARC [3] (Surface Hopping including ARbitary Couplings) which allows to simulate the dynamics of molecular systems after photoexcitation will be outlined to discuss the demands in terms of CPU-time and memory requirements of the underlying ab initio quantum chemistry. The quest for an optimal system architecture in order to run these kinds of calculations will be discussed, focusing on the needs of single core floating point performance as well as memory and I/O requirements. Performance comparisons between the VSC-3 and other types of cluster system with different memory/core ratio and different Infiniband architectures will be presented [4].

The second part of the presentation will focus on quantum mechanical (QM), classical molecular mechanics (MM), and hybrid QM/MM (combined quantum mechanical/molecular mechanical) calculations using GPUs. The performance of readily available, GPU enabled quantum chemical packages like Terachem and molecular mechanics packages like Amber on different kinds of NVIDIA GPUs will be discussed and compared with the corresponding calculations using standard program packages running on traditional, CPU based clusters like the VSC-3 [5].

The contribution will conclude with an outlook for optimal system architecture of VSC-4 and successors from a theoretical chemist’s point of view.

References

Elucidating structural features of rigid spin crossover coordination polymers using HPC

Christian Knoll, Danny Müller, Marco Seifried, and Peter Weinberger

Institute of Applied Synthetic Chemistry, TU Wien

Spin crossover compounds might play an important role for future technological applications, such as memory, sensing or switching devices. An essential feature of such compounds for a potential use in devices is cooperativity between the metal centers to enable an abrupt spin transition [1].

Gaussian 09 Rev.D implemented on VSC-3 was used for quantum chemical structural investigations of iron(II) tetrazole based spin crossover polymers. The aim of this work is a modeling of structural features, so far inaccessible to elucidate with available experimental techniques.

From different measurements it is known, that solvent molecules are located inside the cavities formed by the benzene rings. Due to crystal quality it was not possible to assign unambiguously residual electron density, found on single crystal X-ray analysis, to a defined solvent position. Therefore, a theoretical attempt was chosen to investigate possible preferred solvent positions. This was done by placing a pre-optimized molecule of propylene carbonate into the center of the void of the prior calculated low spin structure. After optimization to a minimum, proven by vibrational analysis, which showed no imaginary frequencies, only one preferred position, as depicted in Figure 1 was obtained. This was even true, if the calculation was started from different starting positions of the solvent guest. The five membered carbonate ring aligns nearly parallel to one of the benzene rings with a centroid distance of 4.14 Å. The carbonyl oxygen has a distance of 3.15 Å towards the N3 of the nearest tetrazole ring. The methyl group of the solvent molecule has a minimum distance of about 2.9 Å towards the N2 of the opposite tetrazole ring, the oxygen is facing to. The central carbon atom lies on the virtual axis connecting the iron(II) centers. The attempt to force a second molecule of propylene carbonate into the void results in convergence failure, as one of the molecules is forced to leave the cavity. These results were reproducible for other solvents as acetonitrile and nitromethane.

Fig. 1: Calculated position of propylene carbonate molecule in the cavity of the spin crossover coordination polymer.

References

Highly parallel simulations of the dynamics of non-Newtonian fluids on GPUs

David Toneian\textsuperscript{a}, Gerhard Kahl\textsuperscript{a}, Gerhard Gompper\textsuperscript{b}, and Roland G. Winkler\textsuperscript{b}

\textsuperscript{a}Institute for Theoretical Physics, TU Wien
\textsuperscript{b}ICS-2 and IAS-2, Forschungszentrum Jülich, Germany

Multi-Scale Problems: In the field of soft matter physics, one is often confronted with systems that contain bodies of vastly different sizes and masses. In a dispersion, for example, there are water molecules with a radius of about 0.3 nm, and then there are solutes, with radii on the order of, say, 10 µm. The separation of length, mass, and time scales makes a physically accurate simulation (e.g. molecular dynamics) infeasible for all but the smallest systems. Some phenomena, however, rely crucially on the effects of the solvent-mediated hydrodynamic interactions between solutes, so that the microscopic degrees of freedom cannot be neglected altogether, giving rise to the need for mesoscopic simulation techniques.

Multi-Particle Collision Dynamics (MPC) is such a simulation algorithm [1, 2]. It models the system as a collection of point-like particles, each of which either represents a mesoscopic volume of the solvent, or a (fraction of) a solute particle. MPC’s design is inherently parallel: in the streaming step, the MPC particles undergo ballistic motion; as such, every MPC particle (of which there can be as many as $10^9$) can be propagated independently. The subsequent collision step introduces an interaction mechanism that allows the MPC particles to exchange momenta. The collision causes interaction only between MPC particles that reside within a common subspace (cell) of the simulation volume, each cell containing only about 10 MPC particles, so that there are still a large number of tasks that can be computed independently.

This high degree of parallelism allows for very straight-forward and efficient implementations of MPC on Graphics Processing Units (GPUs), which results in a reduction of wall-clock time needed for simulations by an order of magnitude, compared to execution on traditional CPUs.

Viscoelastic Fluids: In the original formulation of MPC, the simulated fluid is purely viscous and exhibits Navier-Stokes-type behavior. If, however, one is interested in non-Newtonian fluids, as they are common in technology and biology, one has to introduce elastic degrees of freedom into the system [3, 4].

The possibility explored here is that of linking $N$ MPC particles via harmonic potentials to form linear chains, called polymers, and propagate these polymers (rather than the individual MPC particles) independently, thus retaining high computational efficiency. Non-Newtonian behavior can be observed, specifically in a characteristic quantity called the velocity autocorrelation function in Fourier space, which is measured in the simulations and compared to a closed expression derived analytically using a mix of discrete and continuum mechanics. The theoretical prediction is found to model the obtained data both qualitatively and qualitatively to a highly satisfactory degree [5].

References
VSC School Project:
Parallel software suite for neural network potentials for materials simulations

Andreas Singraber and Christoph Dellago
Computational Physics, University of Vienna

In computational materials science the choice of an appropriate potential energy model is crucial to accurately predict the properties of materials. While sophisticated ab initio methods can be used to calculate energies and forces with high accuracy, their considerable computational cost limits their application to relatively small systems and makes large-scale molecular dynamics (MD) simulations impossible in most cases. Empirical potentials, on the other hand, are computationally far less demanding but also lack in accuracy, particularly if covalent bonds are broken or formed. An alternative approach put forward recently consists in training artificial neural networks (NN) to predict the energies and forces [1]. This new method provides the accuracy of first-principles calculations at a fraction of their computational cost.

In cooperation with Jörg Behler from the Ruhr-Universität Bochum, we are currently developing NN potentials for chalcocite and bulk water with RuNNer, an essentially sequential code that does not yet utilize the capabilites of modern HPC systems. In order to overcome existing limitations this VSC School Project aims at the development of a complete software package that allows users to easily and efficiently generate NN potentials and benefit from their features in large-scale MD simulations. With the previously existing software in mind we identified three main objectives: (1) the development and implementation of a parallel NN training algorithm, (2) the integration of the NN potential method in a well-established MD software package, and (3) the collection and documentation of a consistent set of tools to help with NN potential generation.

In this talk we present our recent efforts to develop a parallelizable training algorithm based on the multi-stream extended Kalman filter [2] and discuss the performance of our MPI implementation on the VSC clusters. Compared to the predecessor code the new procedure exploiting distributed memory allows for a significantly increased size of training data sets and may even further improve fitting results via the incorporation of multiple information in each training step. In addition, we show speed tests and details of our implementation of the NN potential in the popular MD software package LAMMPS [3]. The spatial-decomposition algorithm of LAMMPS works very well together with the NN potential and allows massively parallelized MD simulations on hundreds of cores (see figure). Furthermore, we present our recently developed tool to enhance existing data sets efficiently via MPI4Py and comment on our future strategy to create a user-friendly software package.

References
Microbes represent the most diverse and most abundant group of living organisms. Our knowledge about the biology of prokaryotic microorganisms is mainly obtained from a small minority of lab-cultivable species. Whole-genome sequencing during the last 20 years has pushed this field substantially, and has helped to establish molecular models of microbial lifestyles in even extreme habitats. Applications of this research include fundamental topics such as human health and nutrition, climate change and sustainable use of global resources. Novel technologies, such as metagenomics and single-cell genomics, are currently extending the scope of genomics towards the majority of uncultivable species. These methods rely on sophisticated computational approaches for assembly, binning and annotation of microbial genomes.

**Fig. 1:** Multiple coverage binning. Multiple samples of the same microbiome are taken, e.g. at different time points. The extracted DNA is sequenced and assembled into contigs. For each contig, the coverage in each sample is estimated by mapping of the reads from each sample. The coverage profiles and sequence composition statistics data are clustered. Contigs having similar coverage profiles and similar sequence composition are thereby grouped into the same bin.

**Fig. 2:** Latent strain analysis. Multiple samples of the same microbiome are taken, e.g. at different time points. The extracted DNA is sequenced. k-mers from all sequence reads are hashed. Singular value decomposition (SVD) of the k-mer abundance matrix defines a set of eigengenomes. k-mers are clustered across eigengenomes, and each read is partitioned based on the intersection of its k-mers with each of these clusters. Each partition contains a small fraction of the original data and can be analyzed independently of all others.
The utilization of sequencing coverage, in combination with the application of powerful NGS methods, enabled the most remarkable improvement in the metagenomic binning of assemblies. By evaluating multiple sequencing runs, e.g. from time series, spatial samplings or just different DNA extractions, a vector of relative sequencing coverages is generated for each contig or scaffold. Sequences that originate from the same genome are grouped into bins by the similarity of their coverage vectors (Fig. 1). Latent strain analysis (LSA) is a novel method for grouping short-read sequencing data from many samples into “eigengenome” partitions. This de novo pre-assembly strategy makes it possible to assemble closely related strains present in even very large metagenome data sets (Fig. 2).

The quickly growing number of automatically assembled, near-complete genomes have substantial implications for genome databases, comparative genomics and systems biology of microbes. One of the emerging bioinformatic problems in microbial genomics is the prediction of biological phenotypes and ecological roles of uncharacterized microbial species from their partial genome sequences. The representation of microbial genomes by their protein-coding genes, associated to orthologous or homologous groups, is the most widely used approach for the organization of large-scale genomic data. However, it is computationally very expensive. Comparative genomics for millions of genomes will therefore substantially challenge the existing resources, such as SIMAP [1] and eggNOG [2].

The accessibility of almost complete genome sequences of uncultivable microbial species also necessitates computational methods predicting microbial phenotypes solely based on genomic data. We have recently investigated how comparative genomics can be utilized for the prediction of microbial phenotypes. The PICA framework facilitates application and comparison of different machine learning techniques for phenotypic trait prediction. We have improved and extended PICA’s support vector machine plug-in [3]. In this study, we have put particular emphasis on incomplete genomes and vastly increasing data amounts. We could demonstrate the stability of the predictive power for phenotypic traits by reproducing earlier results, indicating that this method is not perturbed by the rapid growth of genome databases. A new software tool was developed that facilitates the in-depth analysis of phenotype models. It allows associating expected and unexpected protein functions with particular traits. Most of the traits can be reliably predicted in only 60-70% complete genomes, which allows reasonable predictions in genome bins from metagenomes.

We have also established a new phenotypic model that predicts intracellular microorganisms. Thereby we could demonstrate that also independently evolved phenotypic traits, characterized by genome reduction, can be reliably predicted based on comparative genomics. This model is an example of a trait that cannot be associated to single functional marker genes. The predictive power of its model therefore arises from the combination of multiple (mainly absence) genotypic signals. Currently ongoing work indicates very good performance of phenotypic trait prediction also for further, ecologically important traits, as soon as sufficient training data are available. Although these models recover known functional markers, they substantially extend the marker concept by associating many further genes to the phenotypic traits.

References

IQ-TREE HPC: parallelized software for phylogenetic inference

Lam Tung Nguyen, Arndt von Haeseler, and Bui Quang Minh

Center for Integrative Bioinformatics Vienna,
Max F. Perutz Laboratories, University of Vienna and Medical University of Vienna

IQ-TREE [1], the successor of the well known TREE-PUZZLE program [2], is an efficient software to infer phylogenetic trees from sequence data. Based on the maximum likelihood principle, IQ-TREE implemented a stochastic algorithm to explore the tree space, a phylogenomic approach for genome-wide data, an ultrafast bootstrap approximation to assess tree reliability [3], and a fast model selection procedure. More recently, we have actively developed IQ-TREE to analyze big phylogenetic data with thousands of sequences and millions of characters.

Here, we introduce IQ-TREE HPC, a highly parallelized version of IQ-TREE to analyze big data. To this end, we implement two parallelization approaches: a fine-grained and a coarse-grained scheme. In the fine-grained approach, we utilize OpenMP to distribute the likelihood computations among many cores. In the coarse-grained approach, we utilize the message passing interface (MPI) to parallelize the tree search algorithm onto many CPUs. The CPUs only exchange trees sporadically, thus requiring very little communication overhead.

Analysis with the OpenMP version on a big data set (39 protein sequences of 1.8 million amino acids) reduced the computation time from 31 hours (1 core) to 5.5 hours using 12 cores (speedup factor of 5.7). The MPI version showed almost linear speedups for all tested data. Although the OpenMP version does not scale as well as the MPI parallelization, its memory consumption is the same as the sequential version, whereas the memory requirement of the MPI version increases proportionally with the number of CPUs. For example, the aforementioned data set required 20GB RAM for a sequential run. Thus, an MPI version cannot use more than 3 processes per node of 64 GB RAM. Therefore, we are planning to develop a hybrid parallel version (OpenMP+MPI) to best balance the trade off between computation time and memory consumption.

The computational results presented have been achieved using the Vienna Scientific Cluster 3.

References

Ancestral and recent population genomics

Carolin Kosiol\textsuperscript{a}, Dominik Schrempf\textsuperscript{a,b}, Ágnes Jónás\textsuperscript{a,b}, and Christian Schlötterer\textsuperscript{a}

\textsuperscript{a} Institute of Population Genetics, Vetmeduni Vienna
\textsuperscript{b} Vienna Graduate School of Population Genetics

This work focuses on the development of computational methods to investigate the process of adaptation at different time-scales ranging from a few generations in experimental evolution data to phylogenetic analysis of multiple species diverged millions of years ago. In particular, we are interested in DNA sequence evolution by mutation and shifts of allele frequencies.

For long time-scales (Fig. 1), we have developed an approach called POlymorphisms-aware phylogenetic MOdel (PoMo) \cite{1}. It is a phylogenetic Markov model with states representing fixed alleles as well as polymorphisms at different allele frequencies. A substitution is hereby modeled through a mutational event followed by a gradual fixation. Polymorphisms can either be observed in the present (tips of the phylogeny) or be ancestral (present at inner nodes). With this approach, we naturally account for incomplete lineage sorting and shared ancestral polymorphisms (Fig. 1). Our method can accurately and time-efficiently estimate the parameters describing evolutionary patterns for phylogenetic trees of any shape (species trees, population trees, or any combination of those) \cite{1}. Recently, we have implemented our PoMo approach into a tree software package called IQ-tree \cite{2} such that it is fully parallelizable.

For short time scales (Fig. 2), we monitor the allele frequency changes in populations that undergo a selection experiment for temperature adaptation by using next generation sequencing technologies (NGS). The substantial decrease in costs has made it feasible to not only sequence the last generation of a population at the end of a long-term artificial selection experiment but to sequence intermediate generations. The resulting data represents evolutionary trajectories, time-series data, that we analyse using Gaussian Process (GP) models \cite{3}. An advantage of the GP approach is that it can handle replicate population to identify trends across populations. Our experimental design studies the thermal adaptation of two Drosophila species, Melanogaster and Simulans. From inbred lines, the Schlötterer group created ten populations for each of the species. Five populations per species are then kept under cold and hot environments respectively. The experiment is now running for over five years. We will present results of applying our GP software to real data from our close collaboration with the Schlötterer group as well as our plans to incorporate mechanistic models of allele frequencies changes such as PoMo into the GP framework.

The computational results presented have partly been achieved using the Vienna Scientific Cluster (VSC).

References


## List of participants

<table>
<thead>
<tr>
<th>Name</th>
<th>Institution</th>
<th>Email</th>
</tr>
</thead>
<tbody>
<tr>
<td>Michael Alexander</td>
<td>VSC, ZID, TU Wien</td>
<td><a href="mailto:michael.alexander@tuwien.ac.at">michael.alexander@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Michael Aspetsberger</td>
<td>Catalysts GmbH</td>
<td><a href="mailto:michael.aspetsberger@catalysts.cc">michael.aspetsberger@catalysts.cc</a></td>
</tr>
<tr>
<td>Claudia Blaas-Schenner</td>
<td>VSC, ZID, TU Wien</td>
<td><a href="mailto:claudia.blaas-schenner@tuwien.ac.at">claudia.blaas-schenner@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Peter Blaha</td>
<td>TU Wien</td>
<td><a href="mailto:pblaaha@theochem.tuwien.ac.at">pblaaha@theochem.tuwien.ac.at</a></td>
</tr>
<tr>
<td>Ivona Brandič</td>
<td>TU Wien</td>
<td><a href="mailto:ivona@ec.tuwien.ac.at">ivona@ec.tuwien.ac.at</a></td>
</tr>
<tr>
<td>Christian Briese</td>
<td>EODC</td>
<td><a href="mailto:christian.briese@eodc.eu">christian.briese@eodc.eu</a></td>
</tr>
<tr>
<td>Robert Brunthaler</td>
<td>EODC</td>
<td><a href="mailto:robert.brunthaler@eodc.eu">robert.brunthaler@eodc.eu</a></td>
</tr>
<tr>
<td>Minh Quang Bui</td>
<td>CIBIV, University of Vienna</td>
<td><a href="mailto:minh.bui@univie.ac.at">minh.bui@univie.ac.at</a></td>
</tr>
<tr>
<td>Senmao Cao</td>
<td>TU Wien</td>
<td><a href="mailto:senmao.cao@geo.tuwien.ac.at">senmao.cao@geo.tuwien.ac.at</a></td>
</tr>
<tr>
<td>Avtar Cheema</td>
<td>ALLINEO SOFTWARE</td>
<td><a href="mailto:avtar@allinea.com">avtar@allinea.com</a></td>
</tr>
<tr>
<td>Ivan Coluzza</td>
<td>University of Vienna</td>
<td><a href="mailto:ivan.coluzza@univie.ac.at">ivan.coluzza@univie.ac.at</a></td>
</tr>
<tr>
<td>Armin Costa</td>
<td>EURAC Research</td>
<td><a href="mailto:armin.cost@eurac.edu">armin.cost@eurac.edu</a></td>
</tr>
<tr>
<td>Judith Courian</td>
<td>University of Innsbruck</td>
<td><a href="mailto:judith.courian@uibk.ac.at">judith.courian@uibk.ac.at</a></td>
</tr>
<tr>
<td>András Csáki</td>
<td>University of Graz</td>
<td><a href="mailto:andras.csaki@uni-graz.at">andras.csaki@uni-graz.at</a></td>
</tr>
<tr>
<td>Christoph Dellago</td>
<td>University of Vienna</td>
<td><a href="mailto:christoph.dellago@univie.ac.at">christoph.dellago@univie.ac.at</a></td>
</tr>
<tr>
<td>Lukas Einkemmer</td>
<td>University of Innsbruck</td>
<td><a href="mailto:lukas.einkemmer@uibk.ac.at">lukas.einkemmer@uibk.ac.at</a></td>
</tr>
<tr>
<td>Stefano Elefante</td>
<td>TU Wien</td>
<td><a href="mailto:stefano.elefante@geo.tuwien.ac.at">stefano.elefante@geo.tuwien.ac.at</a></td>
</tr>
<tr>
<td>Michael Fink</td>
<td>ZID, University of Innsbruck</td>
<td><a href="mailto:michael.fink@uibk.ac.at">michael.fink@uibk.ac.at</a></td>
</tr>
<tr>
<td>Markus Flatz</td>
<td>University of Salzburg</td>
<td><a href="mailto:mflatz@cosy.sbg.ac.at">mflatz@cosy.sbg.ac.at</a></td>
</tr>
<tr>
<td>Karl Flicker</td>
<td>ZID, Montanuniversität Leoben</td>
<td><a href="mailto:karl.flicker@unileoben.ac.at">karl.flicker@unileoben.ac.at</a></td>
</tr>
<tr>
<td>Karl Fürlinger</td>
<td>LMU München</td>
<td><a href="mailto:karl.fuerlinger@nm.ifi.lmu.de">karl.fuerlinger@nm.ifi.lmu.de</a></td>
</tr>
<tr>
<td>Daniel Ganellari</td>
<td>Karl-Franzens-University Graz</td>
<td><a href="mailto:daniel.ganellari@edu.uni-graz.at">daniel.ganellari@edu.uni-graz.at</a></td>
</tr>
<tr>
<td>Andreas Gattringer</td>
<td>University of Vienna</td>
<td><a href="mailto:andreas.gattringer@univie.ac.at">andreas.gattringer@univie.ac.at</a></td>
</tr>
<tr>
<td>Florian Goldenberg</td>
<td>University of Vienna</td>
<td><a href="mailto:florian.goldenberg@univie.ac.at">florian.goldenberg@univie.ac.at</a></td>
</tr>
<tr>
<td>Patrik Gunacker</td>
<td>TU Wien</td>
<td><a href="mailto:patrik.gunacker@tuwien.ac.at">patrik.gunacker@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Gundolf Haase</td>
<td>University of Graz</td>
<td><a href="mailto:gundolf.haase@uni-graz.at">gundolf.haase@uni-graz.at</a></td>
</tr>
<tr>
<td>Michael Harasek</td>
<td>TU Wien</td>
<td><a href="mailto:michael.harasek@tuwien.ac.at">michael.harasek@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Ernst Haunschmid</td>
<td>VSC, ZID, TU Wien</td>
<td><a href="mailto:ernst.haunschmid@tuwien.ac.at">ernst.haunschmid@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Bernhard Hermann</td>
<td>VSC, ZID, TU Wien</td>
<td><a href="mailto:bernhard.hermann@tuwien.ac.at">bernhard.hermann@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Siegfried Höfinger</td>
<td>VSC, ZID, TU Wien</td>
<td><a href="mailto:siegfried.hoefinger@tuwien.ac.at">siegfried.hoefinger@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Oliver Hofmann</td>
<td>TU Graz</td>
<td><a href="mailto:o.hofmann@tugraz.at">o.hofmann@tugraz.at</a></td>
</tr>
<tr>
<td>Stefan Hummel</td>
<td>econocom-osiatis</td>
<td><a href="mailto:stefan.f.hummel@icloud.com">stefan.f.hummel@icloud.com</a></td>
</tr>
<tr>
<td>Sascha Hunold</td>
<td>TU Wien</td>
<td><a href="mailto:sascha.hunold@tuwien.ac.at">sascha.hunold@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Christian Jordan</td>
<td>TU Wien</td>
<td><a href="mailto:christian.jordan@tuwien.ac.at">christian.jordan@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Dominik Kaaser</td>
<td>University of Salzburg</td>
<td><a href="mailto:dominik@kaaser.at">dominik@kaaser.at</a></td>
</tr>
<tr>
<td>Gerhard Kahl</td>
<td>TU Wien</td>
<td><a href="mailto:gerhard.kahl@tuwien.ac.at">gerhard.kahl@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Christian Knoll</td>
<td>TU Wien</td>
<td><a href="mailto:christian.knoll@tuwien.ac.at">christian.knoll@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Carolin Kosiol</td>
<td>Vetmeduni Vienna</td>
<td><a href="mailto:carolin.kosiol@vetmeduni.ac.at">carolin.kosiol@vetmeduni.ac.at</a></td>
</tr>
<tr>
<td>Christian Kracher</td>
<td>VSC, ZID, University of Vienna</td>
<td><a href="mailto:christian.kracher@univie.ac.at">christian.kracher@univie.ac.at</a></td>
</tr>
<tr>
<td>Sabine Kreidl</td>
<td>ZID, University of Innsbruck</td>
<td><a href="mailto:sabine.kreidl@uibk.ac.at">sabine.kreidl@uibk.ac.at</a></td>
</tr>
<tr>
<td>Dieter Kvasnicka</td>
<td>VSC, ZID, TU Wien</td>
<td><a href="mailto:dieter.kvasnicka@tuwien.ac.at">dieter.kvasnicka@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Matthias Langer</td>
<td>ZID, ZAMG</td>
<td><a href="mailto:matthias.langer@zamg.ac.at">matthias.langer@zamg.ac.at</a></td>
</tr>
<tr>
<td>Dietrich Liko</td>
<td>HEPHY, ÖAW</td>
<td><a href="mailto:dietrich.liko@oeaw.ac.at">dietrich.liko@oeaw.ac.at</a></td>
</tr>
<tr>
<td>Udo Linauer</td>
<td>ZID, TU Wien</td>
<td><a href="mailto:udo.linauer@tuwien.ac.at">udo.linauer@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Name</td>
<td>Institution</td>
<td>Email</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>Peter Marksteiner</td>
<td>VSC</td>
<td><a href="mailto:peter.marksteiner@medizinkraft.com">peter.marksteiner@medizinkraft.com</a></td>
</tr>
<tr>
<td>Andreas Morhammer</td>
<td>TU Wien</td>
<td><a href="mailto:morhammer@iue.tuwien.ac.at">morhammer@iue.tuwien.ac.at</a></td>
</tr>
<tr>
<td>Federico Municchi</td>
<td>TU Graz</td>
<td><a href="mailto:fmunicchi@tugraz.at">fmunicchi@tugraz.at</a></td>
</tr>
<tr>
<td>Aurel Neic</td>
<td>Medical University of Graz</td>
<td><a href="mailto:aurel.neic@medunigraz.at">aurel.neic@medunigraz.at</a></td>
</tr>
<tr>
<td>Francesca Nerattini</td>
<td>University of Vienna</td>
<td><a href="mailto:francesca.nerattini@gmail.com">francesca.nerattini@gmail.com</a></td>
</tr>
<tr>
<td>Lam-Tung Nguyen</td>
<td>CIBIV, University of Vienna</td>
<td><a href="mailto:tung.nguyen@univie.ac.at">tung.nguyen@univie.ac.at</a></td>
</tr>
<tr>
<td>Markus Oppel</td>
<td>University of Vienna</td>
<td><a href="mailto:markus.oppel@univie.ac.at">markus.oppel@univie.ac.at</a></td>
</tr>
<tr>
<td>Alexander Ostermann</td>
<td>University of Innsbruck</td>
<td><a href="mailto:alexander.ostermann@uibk.ac.at">alexander.ostermann@uibk.ac.at</a></td>
</tr>
<tr>
<td>Helmut Pedit</td>
<td>ZID, University of Innsbruck</td>
<td><a href="mailto:helmut.pedit@uibk.ac.at">helmut.pedit@uibk.ac.at</a></td>
</tr>
<tr>
<td>Gernot Plank</td>
<td>Medical University of Graz</td>
<td><a href="mailto:gernot.plank@medunigraz.at">gernot.plank@medunigraz.at</a></td>
</tr>
<tr>
<td>Felix Plasser</td>
<td>University of Vienna</td>
<td><a href="mailto:felix.plasser@univie.ac.at">felix.plasser@univie.ac.at</a></td>
</tr>
<tr>
<td>Raimund Podlouky</td>
<td>University of Vienna</td>
<td><a href="mailto:raimund.podlouky@univie.ac.at">raimund.podlouky@univie.ac.at</a></td>
</tr>
<tr>
<td>Martin Pöll</td>
<td>ZID, University of Innsbruck</td>
<td><a href="mailto:martin.poell@uibk.ac.at">martin.poell@uibk.ac.at</a></td>
</tr>
<tr>
<td>Thomas Ponweiser</td>
<td>RISC Software GmbH</td>
<td><a href="mailto:thomas.ponweiser@risc-software.at">thomas.ponweiser@risc-software.at</a></td>
</tr>
<tr>
<td>Martina Prugger</td>
<td>University of Innsbruck</td>
<td><a href="mailto:martina.prugger@uibk.ac.at">martina.prugger@uibk.ac.at</a></td>
</tr>
<tr>
<td>Christian Rab</td>
<td>University of Vienna</td>
<td><a href="mailto:christian.rab@univie.ac.at">christian.rab@univie.ac.at</a></td>
</tr>
<tr>
<td>Thomas Rattei</td>
<td>University of Vienna</td>
<td><a href="mailto:thomas.rattei@univie.ac.at">thomas.rattei@univie.ac.at</a></td>
</tr>
<tr>
<td>Josef Redinger</td>
<td>TU Wien</td>
<td><a href="mailto:josef.redinger@tuwien.ac.at">josef.redinger@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Irene Reichl</td>
<td>VSC, ZID, TU Wien</td>
<td><a href="mailto:irene.reichl@tuwien.ac.at">irene.reichl@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Lorenz Romaner</td>
<td>Materials Center Leoben</td>
<td><a href="mailto:lorenz.romaner@mcl.at">lorenz.romaner@mcl.at</a></td>
</tr>
<tr>
<td>Ulrich Rüde</td>
<td>FAU Erlangen-Nürnberg</td>
<td><a href="mailto:ulrich.ruete@fau.de">ulrich.ruete@fau.de</a></td>
</tr>
<tr>
<td>Thomas Ruh</td>
<td>TU Wien</td>
<td><a href="mailto:thomas.ruh@tuwien.ac.at">thomas.ruh@tuwien.ac.at</a></td>
</tr>
<tr>
<td>Karl Rupp</td>
<td>TU Wien</td>
<td><a href="mailto:me@karlrupp.net">me@karlrupp.net</a></td>
</tr>
<tr>
<td>Alois Schögl</td>
<td>IST Austria</td>
<td><a href="mailto:alois.schoegl@ist.ac.at">alois.schoegl@ist.ac.at</a></td>
</tr>
<tr>
<td>Petra Seibert</td>
<td>BOKU Vienna</td>
<td><a href="mailto:petra.seibert@univie.ac.at">petra.seibert@univie.ac.at</a></td>
</tr>
<tr>
<td>Samuel Senoner</td>
<td>EURAC Research</td>
<td><a href="mailto:samuel.senoner@eurac.edu">samuel.senoner@eurac.edu</a></td>
</tr>
<tr>
<td>Andreas Singraber</td>
<td>University of Vienna</td>
<td><a href="mailto:andreas.singraber@univie.ac.at">andreas.singraber@univie.ac.at</a></td>
</tr>
<tr>
<td>Alexander Spinn</td>
<td>University of Innsbruck</td>
<td><a href="mailto:alexander.spinn@uibk.ac.at">alexander.spinn@uibk.ac.at</a></td>
</tr>
<tr>
<td>Stephan Stadlbauer</td>
<td>IST Austria</td>
<td><a href="mailto:stephan.stadlbauer@ist.ac.at">stephan.stadlbauer@ist.ac.at</a></td>
</tr>
<tr>
<td>Manfred Stepponat</td>
<td>ZID, TU Graz</td>
<td><a href="mailto:stepponat@tugraz.at">stepponat@tugraz.at</a></td>
</tr>
<tr>
<td>Claudia Stocits</td>
<td>Medical University of Vienna</td>
<td><a href="mailto:claudia.stocits@meduniwien.ac.at">claudia.stocits@meduniwien.ac.at</a></td>
</tr>
<tr>
<td>Herbert Störi</td>
<td>VSC, TU Wien</td>
<td><a href="mailto:stoeri@iap.tuwien.ac.at">stoeri@iap.tuwien.ac.at</a></td>
</tr>
<tr>
<td>Markus Stöhr</td>
<td>VSC, ZID, BOKU Vienna</td>
<td><a href="mailto:markus.stoehr@boku.ac.at">markus.stoehr@boku.ac.at</a></td>
</tr>
<tr>
<td>Toma Susi</td>
<td>University of Vienna</td>
<td><a href="mailto:toma.susi@univie.ac.at">toma.susi@univie.ac.at</a></td>
</tr>
<tr>
<td>Martin Thaler</td>
<td>ZID, University of Innsbruck</td>
<td><a href="mailto:martin.thaler@uibk.ac.at">martin.thaler@uibk.ac.at</a></td>
</tr>
<tr>
<td>David Toneian</td>
<td>TU Wien</td>
<td><a href="mailto:david.toneian@gmail.com">david.toneian@gmail.com</a></td>
</tr>
<tr>
<td>Luca Tubiana</td>
<td>University of Vienna</td>
<td><a href="mailto:luca.tubiana@univie.ac.at">luca.tubiana@univie.ac.at</a></td>
</tr>
<tr>
<td>Marián Vajtersic</td>
<td>University of Salzburg</td>
<td><a href="mailto:marian@cosy.sbg.ac.at">marian@cosy.sbg.ac.at</a></td>
</tr>
<tr>
<td>Isabella Weger</td>
<td>ECMWF</td>
<td><a href="mailto:isabella.weger@ecmwf.int">isabella.weger@ecmwf.int</a></td>
</tr>
<tr>
<td>Josef Weinhub</td>
<td>TU Wien</td>
<td><a href="mailto:weinhub@iue.tuwien.ac.at">weinhub@iue.tuwien.ac.at</a></td>
</tr>
<tr>
<td>Matthias Wiesenberger</td>
<td>University of Innsbruck</td>
<td><a href="mailto:matthias.wiesenberger@uibk.ac.at">matthias.wiesenberger@uibk.ac.at</a></td>
</tr>
<tr>
<td>Ursula Winkler</td>
<td>ZID, Karl-Franzens-University Graz</td>
<td><a href="mailto:ursula.winkler@uni-graz.at">ursula.winkler@uni-graz.at</a></td>
</tr>
<tr>
<td>Jan Zabloudil</td>
<td>VSC, ZID, BOKU Vienna</td>
<td><a href="mailto:jan.zabloudil@tuwien.ac.at">jan.zabloudil@tuwien.ac.at</a></td>
</tr>
</tbody>
</table>
Author index

Akbari, Hoda, 44
Alexander, Michael, 41
Aspetsberger, Michael, 40
Augustin, Christoph, 1
Bancelin, David, 26
Benedict, Petra, 44
Blaha, Peter, 32
Brandic, Ivona, 42
Briese, Christian, 9
Bui, Minh Quang, 56
Cao, Senmao, 10
Carpen-Amarie, Alexandra, 45
Coluzza, Ivan, 27
Crozier, Andrew, 1
Csaki, Andras, 8
Dellago, Christoph, 52
Djurdjic, Aleksandar, 40
Ebner, Wolfgang, 40
Einkemmer, Lukas, 20, 21
Elefante, Stefano, 10
Elsässer, Robert, 44
Fürlinger, Karl, 22
Federspiel, Christian, 40
Flatz, Markus, 46
Fuchs, Tobias, 22
Ganellari, Daniel, 5
Gerstmayr, Andreas, 40
Gompper, Gerhard, 51
Goniva, Christoph, 19
González, Leticia, 48, 49
Gumacker, Patrik, 36
Höfinger, Siegfried, 14
Haase, Gundolf, 1, 4, 5
Haddadi, Bahram, 18
Haeseler, Arndt von, 56
Harasek, Michael, 18
Hasenauer, Stefan, 9
Haunschmid, Ernst, 14
Hausel, Andreas, 36
Held, Karsten, 36
Held, Markus, 25
Hofmann, Oliver T., 34
Hunold, Sascha, 45
Jónás, Ágnes, 57
Johnston, Colin, 26
Jordan, Christian, 18
Kaaser, Dominik, 44
Kahl, Gerhard, 51
Karabelas, Elias, 1
Karch, Rudolf, 28
Kendl, Alexander, 25
Kidd, Richard, 9
Knoll, Christian, 50
Kosiol, Carolin, 57
Kowalewski, Roger, 22
Kreidl, Sabine, 13
Liebmann, Manfred, 4
Liko, Dietrich, 39
Mücke, Werner, 9
Müller, Danny, 50
Maindl, Thomas I., 26
Marksteiner, Peter, 14
Mayr-Schmölzer, Wernfried, 47
Mistelbauer, Thomas, 9
Mittendorfer, Florian, 47
Morhammer, Andreas, 23
Municchi, Federico, 19
Naeimi, Vahid, 10
Neic, Aurel, 1, 4, 5
Nerattini, Francesca, 27
Nguyen, Lam Tung, 56
Oppel, Markus, 48, 49
Ostermann, Alexander, 13, 21
Plank, Gernot, 1, 4
Plasser, Felix, 48
Podloucky, Raimund, 30
Ponweiser, Thomas, 12
Prassl, Anton, 1
Prugger, Martina, 21
Rab, Christian, 26
Radl, Stefan, 19
Rattei, Thomas, 54
Razumovskiy, Vsevolod, 33
Redinger, Josef, 47
Reichl, Irene, 14
Ribarics, Reiner, 28
Romaner, Lorenz, 33
Rüde, Ulrich, 16
Rudolf, Florian, 23
Ruh, Thomas, 32
Rupp, Karl, 23, 24
Sangiovanni, Giorgio, 36
Schäfer, Christian, 26
Scheibe, Daniel, 33
Schötteler, Christian, 57
Schlägl, Alois, 37
Schreiner, Wolfgang, 28
Schrepf, Dominik, 57
Seifried, Marco, 50
Senoner, Samuel, 38
Singraber, Andreas, 52
Spitaler, Jürgen, 33
Stöhr, Markus, 14
Störi, Herbert, 14
Stadlbauer, Stephan, 37
Steyerleithner, Patrick, 26
Stocsits, Claudia, 28
Strigl, Martin, 40
Susi, Toma, 35
Toneian, David, 51
Träff, Jesper Larsson, 45
Truhetz, Heimo, 8
Tubiana, Luca, 27
Vajtersic, Marián, 46
Vorobyov, Eduard I., 26
Wagner, Wolfgang, 9, 10
Wallerberger, Markus, 36
Weger, Isabella, 6
Weinberger, Peter, 50
Weinbub, Josef, 23, 24
Wiesenberger, Matthias, 25
Winkler, Roland G., 51
Zojer, Egbert, 34
Links with information about the Vienna Scientific Cluster

- VSC homepage with general information:
  http://vsc.ac.at
- Overview of past and current projects on VSC:
  http://vsc.ac.at/research/projects-on-the-vsc
- List of publications resulting from work on VSC:
  http://vsc.ac.at/publications
- Service website for project applications:
  https://service.vsc.ac.at
- Cluster usage statistics:
  VSC-2: https://service.vsc.ac.at/public/statistics/usage/2
  VSC-3: https://service.vsc.ac.at/public/statistics/usage/3
- VSC Wiki:
  https://wiki.vsc.ac.at
- Information about GPUs on VSC:
  https://wiki.vsc.ac.at/doku.php?id=doku:vsc3_gpu
- User support:
  service@vsc.ac.at

Impressum/Imprint

Herausgeber/Editor:
VSC - Vienna Scientific Cluster
Zentraler Informatikdienst der Technischen Universität Wien
Wiedner Hauptstraße 8-10/E020, 1040 Wien
Zentraler Informatikdienst der Universität Wien
Universitätsstraße 7, 1010 Wien
Zentraler Informatikdienst der Universität für Bodenkultur
Peter-Jordan-Straße 82, 1190 Wien
Zentraler Informatikdienst der Technischen Universität Graz
Steyrergasse 30/1, 8010 Graz
Zentraler Informatikdienst der Universität Innsbruck
Technikerstraße 23, A-6020 Innsbruck

Redaktion/Edited by:
Irene Reichl (irene.reichl@tuwien.ac.at)
Claudia Blaas-Schenner (claudia.blaas-schenner@tuwien.ac.at)
Jan Zabloudil (jan.zabloudil@tuwien.ac.at)

Layout:
Irene Reichl, Claudia Blaas-Schenner

Bildnachweis/Photo Credits:
VSC (Cover oben).
MONDI-HOLIDAY Seeblickhotel Grundlsee (Cover unten).
Copyright of pictures within individual contributions by authors.

© Februar 2016