
ASHPC24
Austrian-Slovenian HPC Meeting
Grundlsee, 10–13 June 2024

Austrian-Slovenian HPC Meeting 2024–ASHPC24

Grundlsee, 10–13 June 2024

https://ashpc.eu

Welcome to ASHPC24

A warm welcome to ASHPC24, the Austrian-Slovenian HPC Meeting 2024, which is being held from June 10–13, 2024 at the Seeblickhotel Grundlsee in Austria.

In the rapidly evolving landscape of scientific research, the advent of exascale supercomputers presents immense scientific potential, promising groundbreaking discoveries across various scientific disciplines. However, harnessing this potential involves formidable challenges, including the complexities of the hardware ecosystem and the development of applications that can effectively utilize exascale capabilities. ASHPC24 is an interdisciplinary conference devoted to advances in algorithms, scientific software, and applications to address challenges across diverse domains of computational science and scientific computing. ASHPC24 is the tenth conference in a series that highlights a fruitful collaboration and exchange of ideas between the Austrian and Slovenian HPC communities.

ASHPC24 features an exciting program of 5 keynote lectures, 37 contributed talks, and 14 posters covering a broad range of HPC topics such as weather and climate, quantum computing, artifical inteligence, life sciences, and industrial applications to name but a few.

ASHPC24 is organized by EuroCC Austria and EuroCC Slovenia, the National Competence Centres for Supercomputing, Big Data and Artificial Intelligence, in cooperation with the Vienna Scientific Cluster (VSC), Austria, the Research Area Scientific Computing in Innsbruck, Austria, and the Slovenian consortium for high performance computing (SLING). Financial support is provided by the Austrian HPC Association and by the EuroCC 2 project.

On behalf of the program and organizing committees we wish you an inspiring and productive meeting!

The Program Committee:

Matej Praprotnik (chair): National Institute of Chemistry, Slovenia

Aleš Zamuda: Faculty of Electrical Engineering and Computer Science, University of Maribor, Slovenia
Barbara Krašovec: Jožef Stefan Institute, Slovenia
Claudia Blaas-Schenner: VSC Research Center, TU Wien and EuroCC Austria, Austria
Markus Oppel: Department of Theoretical Chemistry, University of Vienna, Austria

The Organizing Committee:

Eduard Reiter (chair): Research Area Scientific Computing, University of Innsbruck, Austria

Malgorzata Goiser: VSC Research Center, TU Wien and EuroCC Austria, Austria
Urša Vodopivec: ARNES – Academic and Research Network of Slovenia, Slovenia

Schedule

Monday, June 10, 2024

Start		Title
	EuroCC 2 Central European NCCs Workgroup Meeting	
14:00	GET TOGETHER & COFFEE	
14:30	Markus Stöhr Claudia Blaas-Schenner	CE-NCCs-WG Plenary
15:20	BREAK	
15:30	Natascha Trzepizur Julia Wimmer	Pimp my LinkedIn Profile – Workshop
16:45	BREAK	
17:00	Thomas Mayerhofer Simeon Harrison Markus Stöhr	AI Focus

Start		Title
	HPC-Austria Networking Meeting	
14:00	GET TOGETHER & COFFEE	
14:30	Elias Wimmer	MUSICA: Update & Outlook
15:30	Austrian HPC Groups	Networking

Start		Title
	Austrian-Slovenian HPC Meeting 2024–ASHPC24	
18:30	ASHPC24 REGISTRATION	
19:00	ASHPC24 WELCOME DINNER	
20:30	ASHPC24 WELCOME RECEPTION	
20:30	Alexander Ostermann Claudia Blaas-Schenner	10 Years of A(S)HPC Meetings
20:45	Martin Pfister Simeon Harrison	Inauguration of Austria's new supercomputer "VSCrunchy"
20:55	ALL	"VSCrunchy" Guided Tours & Let's Celebrate Together!

Tuesday, June 11, 2024

Start		Title
09:00	Matej Praprotnik	Welcome by the Program Chair
09:07	Eduard Reiter	Welcome by the Organizing Chair
09:15	Roeland ter Hoeven	ParityOS – The operating system for quantum computers (Keynote)
10:00	Janez Povh	Solving the Stable Set problem by D-Wave Quantum Annealer
10:15	Eduarda Sangiogo Gil	Parallelized Implementation of the Coupled-Trajectory Tully Surface Hopping (CT-TSH) Method for Efficient Molecular Simulations
10:30	COFFEE	
11:00	Endri Deliu	Beyond LLMs: Small is the new Large – Towards Pipelines of Modular Models – Implications for HPC Systems and Deployments
11:15	Tina Črnigoj Marc	Innovation in business: HPC as an opportunity for SMEs
11:30	Matic Brank Tina Črnigoj Marc	EXCELLERAT CoE: The European Centre of Excellence for Engineering Applications
11:45	Florian Atzenhofer-Baumgartner	HPC and DHInfra
12:00	Jurica Špoljar	Croatian National Advanced Computing Resources Supek and Vrančić
12:15	BREAK	
12:30	LUNCH	
14:00	Jeanette Nilsson	AI and HPC: Paving the Way for Europe's Digital Future (Keynote)
14:45	David Pfahler	Live resource management in HPC applications: Resilient hybrid computing in IMS' MBMW processing pipeline
14:55	Paul Heistracher	Enhancing Computational Throughput in Multi-Beam Mask Writing: Leveraging Celery as a Distributed Task Queue on Heterogeneous Clusters
15:05	Martin Jurkovič	High-throughput image processing in multi-beam mask writing
15:15	BREAK	
15:30	COFFEE	
16:00	Moreno Guernelli	Leonardo's Role in the Pre-exascale Era
16:15	Teo Prica	Development and supporting activities on EuroHPC Vega
16:30	Dejan Lesjak	Software containers in HPC
16:45	Alja Prah	Overcoming challenges in building scientific software for HPC
17:00	Violeta Šikaleska Samo Lorenčič	Lessons learned/experience from the battlefield: HPC user support

17:15	POSTER LIGHTNING TALKS	
17:35	POSTER SESSION 1	
	Tomas Kozubek	European Master for High Performance Computing (EUMaster4HPC) project
	Jan Jona Javoršek	MaX Centre of Excellence and the EuroHPC ecosystem (after the 1st year of MaX3)
	Matej Praprotnik	MultiXscale – a EuroHPC JU Center of Excellence
	Lubomir Riha	SPACE – Scalable Parallel Astrophysical Codes for Exascale – Center of Excelence
	Tina Črnigoj Marc	EXCELLERAT CoE: Tackling Next-Generation Engineering Challenges in the Exascale Era
	Tina Črnigoj Marc	FF4EuroHPC: SME Innovation Through HPC
	Tomas Karasek	NCC Czechia: HPC World Guide
	Beata Krawczyk-Bryłka	Coopetition as a model of cooperation in the Polish National Competence Center
	Eva Gergely	The Role of Public High Performance Computing Infrastructures in Advancing AI Across European Industries and Academia
	Dominik Brandstetter	Band structure formation in metal-organic nanostructures
	Andreas Lindner	Quantum Vacuum Simulations in the Heisenberg–Euler Weak-Field Expansion
	Michal Kravčenko	Deep learning for scientific data compression
	Ivan Vialov	Some Quirks of Using Shared Memory in CUDA Fortran
	Teo Prica	Hyperparameter Optimization with Differential Evolution on Multiple GPUs
19:00	BREAK	
19:30	DINNER	

Wednesday, June 12, 2024

Start		Title
09:00	Marko Robnik-Šikonja	Building large language models: HPC users' perspective (Keynote)
09:45	Aleš Zamuda	Randomised Optimisation Algorithms in DAPHNE
10:00	Bendikt Hartl	Evolutionary Implications of Multi-Scale Intelligence
10:15	Draško Tomić	Zettaflops Computing: Why, when, and how?
10:30	COFFEE	
11:00	Philipp Gschwandtner	Beyond Benchmarks: Comparing Parallel Programming Models in Real-World Scenarios
11:15	Peter Thoman	Transparently Scaling Applications to Multiple GPUs with Celerity
11:30	Ratko Pilipović	Optimising routines with sparse matrices on FPGA devices
11:45	Ruben Laso	What You Always Wanted To Know About C++ Performance Portability (But Were Afraid to Do)
12:00	Ivona Vasileska	Performance and Portability Assessment of CUDA and SYCL for Particle-in-Cell codes on different GPUs
12:15	BREAK	
12:30	LUNCH	
14:00	Irene Schicker	Advancements, prospects, and challenges in weather and climate prediction: the rise of machine learning (Keynote)
14:45	Aiko Voigt	iCONtainer: climate and weather simulations with ICON in a singularity container
15:00	Mark Edwin Tupas	Enabling Global Flood Monitoring with SAR Datacubes and the Vienna Scientific Cluster
15:15	BREAK	
15:30	COFFEE	
16:00	Elias Wimmer	MUSICA (MUlti-SIte Computer Austria)
16:15	Benjamin Hackl	Cython vs. Numba vs. Mojo: A Comparison of Different Approaches to speedup Python Language Execution
16:30	Alois Schlögl	How much memory per CPU core is requested?
16:45	Wiktor Nastał	Automating Access to OpenID Connect Services from Linux Systems
17:00	James E. McKevitt	Accelerating Fortran Codes: Merging Intel Coarray Fortran with CUDA and OpenMP
17:15	POSTER SESSION 2 (same posters as in session 1)	
19:00	BREAK	
19:30	DINNER	

Thursday, June 13, 2024

Start		Title
09:00	Dejan Štepec	Cell Detection for the 2020s – Computational Pathology and HPC Perspective (Keynote)
09:45	Amaury Coste	Implicit Solvation Machine Learning Model for Molecular Simulations of Ionic Media
10:00	Kirils Surovovs	OpenFOAM CFD simulation performance and scalability analysis
10:15	Thomas Rattei	Accounting and efficient use of GPUs in a slurm based HPC cluster
10:30	COFFEE	
11:00	Wolfgang Husinsky	Molecular Dynamic simulations addressing the missing single collision peak in low energy heavy ion scattering
11:15	Wolfgang Husinsky	NCIS-LA, Nature Computer investigative Simulations of Laser Ablation
11:30	Dieter Kvasnicka	Checkpointing on VSC
11:45	Siegfried Höfinger	Exploring Energy-Efficient GPU Computing
12:00	CLOSING + ASHPC25	
12:15	LUNCH	
14:00	END	

Contents

Keynote Talk:

ParityOS – The operating system for quantum computers

Roeland ter Hoeven

ParityQC and University of Innsbruck

Quantum computing has developed rapidly in the last years, both in terms of the experimental control of quantum systems and the theoretical knowledge of quantum algorithms. Quantum computers have progressed from few-qubit prototypes to small computers that can do non-trivial calculations. In a few cases these calculations get close to the limit of what is currently possible with classical computers.

The ParityQC Architecture [1, 2] provides a fundamentally new way of solving real-world problems on quantum computers, defining both the hardware layout and the algorithms. It is a quantum computing architecture that aims to provide new pathways to work efficiently with every currently available and future quantum computing platform.

ParityOS is an operating system for quantum computers, currently under development by ParityQC. It connects users with real world problems to quantum devices that can be used to solve those problems. ParityOS includes a variety of tools with the compiler [2] at its heart. It translates any problem to a quantum algorithm with tailored hardware designs using the ParityQC Architecture. The goal is to enable hardware developers, end-users, scientists, and quantum computing enthusiasts around the world to access the most efficient way to solve problems on quantum computers with a single function call.

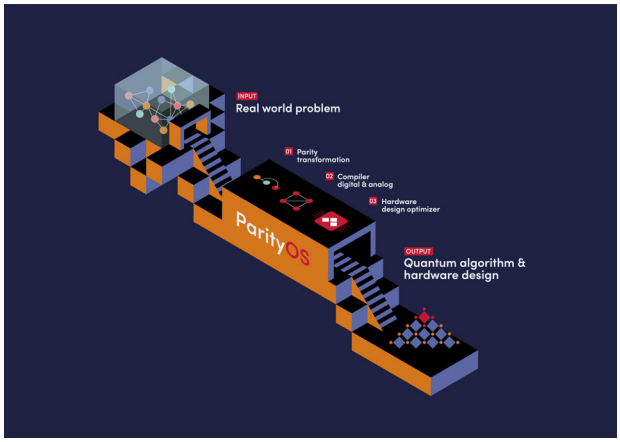


Fig. 1: A schematic overview of ParityOS. The input is a real world problem, for example a combinatorial optimization problem. The parity compilers translate this problem to a quantum algorithm that can be implemented on quantum hardware. This process is compatible with all major quantum computing platforms, and at the same time offers hardware-specific optimizations for the different platforms.

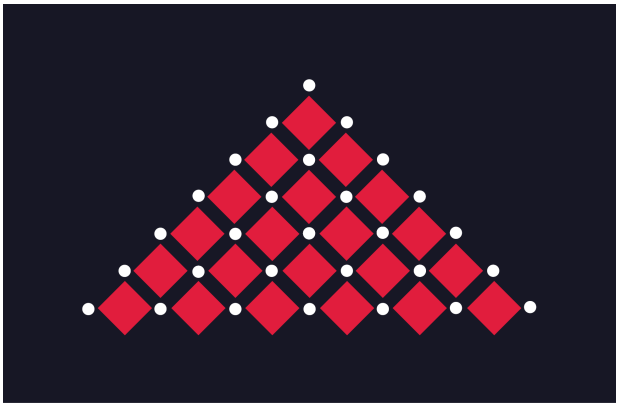


Fig. 2: The simplest version of the ParityQC Architecture. All problem-depending interactions are implemented with local operations on the qubits (white circles). The red squares represent the parity constraints that are a crucial part of the parity transformation. The constraints can be implemented natively on the hardware with four-qubit interactions or decomposed into multiple two-qubit interactions.

References

[1] Lechner, W., Hauke, P., and Zoller, P., Sci. Adv. **1**, e1500838(2015).

[2] Ender, K., ter Hoeven, R., Niehoff, B. E., Drieb-Schön, M., and Lechner, W., Quantum **7**, 950 (2023).

Solving the Stable Set Problem by D-Wave Quantum Annealer

Aljaž Krpan[a], **Dunja Pucher**[b], **and *Janez Povh***[c,d]

[a] *Faculty of Mathematics and Physics, University of Ljubljana, Ljubljana, Slovenia*
[b] *Alpen-Adria-Universität Klagenfurt, Klagenfurt, Austria*
[c] *Rudolfovo Institute, Novo Mesto, Slovenia*
[d] *Faculty of Mechanical Engineering, University of Ljubljana, Ljubljana, Slovenia*

In this talk, we present a comprehensive approach to solve one of the most renowned NP-hard problems: the Stable Set Problem. We reframe it as a Quadratic Unconstrained Binary Optimization (QUBO) problem and tackle it using both the exact solvers like GUROBI or BiqBin, and approximate quantum annealing solvers such as the D-Wave quantum processing unit and D-Wave hybrid solver.

In cases where the capabilities of quantum annealing solvers are exceeded, we introduce a specialized decomposition technique. This method effectively breaks down the input graph into smaller graphs, rendering them more manageable for quantum annealers.

Finally, we conduct a thorough assessment of both quantum solvers alongside a classical simulated annealing solver. Our extensive numerical evaluation on various families of benchmark graphs demonstrates that the simulated annealing approach remains highly competitive. This competitiveness is primarily attributed to its simplicity and computational efficiency compared to quantum solvers. Our results may indicate which solvers are used behind the D-Wave hybrid solver.

References

[1] Povh, J., & Pucher, D, In: DROBNE, Samo (ur.), et al. SOR '23 : proceedings of the 17th International Symposium on Operational Research in Slovenia: Bled, Slovenia, September 20-22 (2023)

[2] Povh, J., & Krpan, A, In: DROBNE, Samo (ur.), et al. SOR '23 : proceedings of the 17th International Symposium on Operational Research in Slovenia: Bled, Slovenia, September 20-22 (2023)

Parallelized Implementation of the Coupled-Trajectory Tully Surface Hopping (CT-TSH) Method for Efficient Molecular Simulations

Eduarda Sangiogo Gil

Institute of Theoretical Chemistry, University of Vienna, Austria

The time evolution of a (non-relativistic) molecular system is determined by the time-dependent Schrödinger equation (TDSE). Unfortunately, solving the TDSE and simulating quantum dynamics in molecules is a challenging problem. One common and effective approach is to treat the nuclear dynamics with classical mechanics and only the electron dynamics with quantum mechanics. This integration of classical and quantum descriptions gives rise to various mixed quantum-classical dynamics methods. Over the last few decades, numerous quantum-classical methods have been proposed, with the Tully Surface Hopping (TSH) method emerging as one of the most popular among them. However, TSH is known to have a systematic problem due to the disconnect between how electrons and nuclei evolve, an issue commonly referred to as "overcoherence". Quantum-classical numerical approaches derived from exact factorization, such as the Coupled-Trajectory Tully Surface Hopping (CT-TSH), appear to enhance the description of quantum decoherence, surpassing, the TSH method. This method allows to recover some quantum effects thanks to the coupling among the trajectories. However, the coupling among the trajectories demands a greater computational effort when contrasted with the independent-trajectory TSH, which is a straightforwardly parallelizable algorithm. In this context, we present the implementation of the CT-TSH method using MPI parallelization. Given the inherent coupled trajectory nature of the algorithm, where the propagation of electronic coefficients in CT-TSH necessitates reconstructing the nuclear density at each time step, synchronized trajectory propagation is required. In this context, a parallelization strategy, such as MPI, becomes *crucial* for comprehensively studying molecules in full dimensionality. Overall, the runtime of our parallelized implementation of CT-TSH is comparable to that of a standard TSH approach, as the majority of calculations are performed in parallel.

Beyond LLMs: Small is the new Large Towards Pipelines of Modular Models Implications for HPC Systems and Deployments

Endri Deliu

University of Vienna and EuroCC Austria, Austria

The advent of foundation models in AI, and specifically of large language models (LLMs), has projected HPC systems, typically employed for training such model types, to visibility and prominence for modern AI development. LLMs however, have by design structural problems when it comes to data quality, hallucination free generation, controllability, safety, as well as energy efficiency. Running a prediction for an LLM, requires to always invoke all of their tens to hundreds of billions of parameters, regardless of the fact that the vast majority of them is not relevant for a particular domain/task. The "bigger is better" philosophy, at the core of current LLM development, the approach of making these models ever larger with more parameters, although seemingly having no end in sight in terms of yielding incremental improvements across many tasks, is however achieved at the back of increasingly infeasible computational budgets. Ultimately, LLMs can be viewed as "monolithic systems", from a software systems perspective with all the downsides that such systems present, among other things at domain adaptation, timely new knowledge acquisition, ease of deployment and operation, and last but not least, controllability: our ability to guide and control their behavior for a desired context. We present an overview of different approaches and techniques, highlighting "modularity" and "composition" as an alternative to very large models. Instead of a single very large model, designing "pipelines" of smaller less powerful and more specialized models to adapt for a domain or a task yields much better cost to performance tradeoffs. Instead of increasing the size of LLMs and have incremental small improvements across the board at the cost of very high computational and energy budgets, we posit smaller more targeted models can achieve large gains in a particular task, when composed within a modular and non-monolithic AI system.

While best practices in designing modular AI systems are still an open question, we'll show some of the tooling and principles emerging around this approach, including a few example architectures. We introduce a few modularity dimensions, via concepts like "*data modularity*", and "*pipeline modularity*" for the creation and deployment of more modular AI systems. In addition, we present the implications that a modular approach has on HPC infrastructure specifically, including infrastructure relevant for training as well as for deployment and inference components. Lastly we'll give a picture of the emerging tools ecosystem.

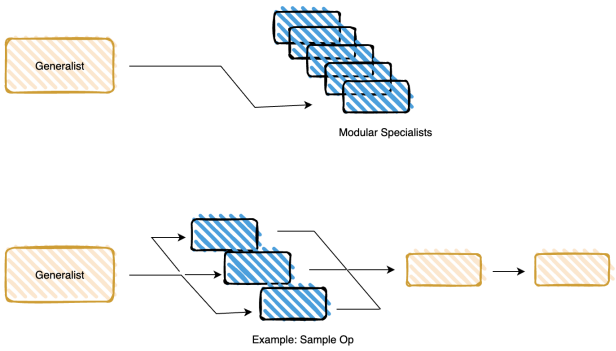


Fig. 1: Data and Pipeline Modularity in Modular and Compositional AI Systems

References

[1] Zaharia, M., et al., The Shift from Models to Compound AI Systems (2024) https://bair.berkeley.edu/blog/2024/02/18/compound-ai-systems/

Innovation in business: HPC as an opportunity for SMEs

Tomi Ilijaš and *Tina Črnigoj Marc*

Arctur d.o.o., Slovenia

High-performance computing (HPC) is a game-changer for businesses looking to stay ahead of the competition. By leveraging cutting-edge technologies such as cloud supercomputing, artificial intelligence (AI), machine learning (ML), and big data analysis, companies can develop products and services with higher added value, paving the way for novel industrial applications. European Small and Medium-sized Enterprises (SMEs) are the backbone of Europe's economy but encounter several challenges in their quest to access HPC resources. These challenges include a lack of awareness of the use of HPC, budget constraints, data security risks, and limited access to HPC infrastructure.

The European High-Performance Computing Joint Undertaking [1] is a pioneering initiative established in 2018 by the European Union to accelerate the development and deployment of HPC technologies across Europe. The EUROHPC JU financial support can be instrumental in driving innovation and competitiveness in European industry. When talking about funding and cascading programs for the industry as well as accessing HPC infrastructure, initiatives such as National Competence Centres (NCCs) could approach the industry, especially SMEs, to help them on their path towards Industry 4.0 and 5.0. 33 NCCs were established within the EUROCC project [2] to become a central point of contact for HPC in respective European countries. NCCs are raising awareness of the HPC uptake and putting a huge effort into promoting the use of HPC to industrial end-users. NCC HPC SLING [3] did a step further and established the HPC4SME AAT Task Force, led by partner Arctur with the support of CASTIEL. In collaboration with 22 NCCs, the HPC4SME Automated Assessment Tool (AAT) was successfully upgraded and translated into 16 languages and is used within the NCCs.

The tool is widely used among European SMEs to discover if and how the organization can benefit from supercomputing technology in R&D and other business processes. NCCs can now access SMEs, support them to fill in complex questionnaires, discuss the valuable report and recommendations SMEs received after submitting the questionnaire, and provide them with further services, based on the report analysis. In this presentation, the HPC4SME AAT Task Force, initial results, and the use of HPC for SMEs will be presented.

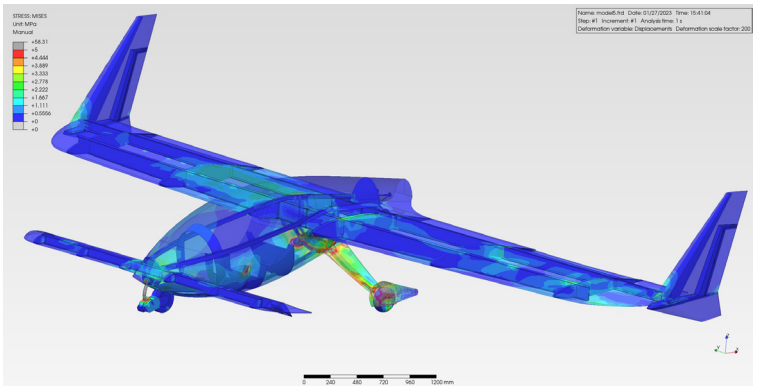


Fig. 1: Success Story: Slovenian SME AFormX utilized HPC4SME AAT and received support from the NCC HPC SLING partner, the University of Maribor. They successfully applied for the FF4EuroHPC Open Call and developed an HPC-based workflow, which they then offered as a service to the aviation market.

References

[1] EUROHPC JU: https://eurohpc-ju.europa.eu/

[2] EUROCC 2: https://www.eurocc-access.eu/

[3] SLING.SI: https://www.sling.si/en/

EXCELLERAT CoE: The European Centre of Excellence for Engineering Applications

Matic Brank[a] **and** ***Tina Črnigoj Marc***[b]

[a] *LECAD, University of Ljubljana, Slovenia*
[b] *Arctur d.o.o., Slovenia*

The use case titled "Engineering design and digital twin of the first wall of a tokamak fusion reactor" will be presented in detail. This use case involves a full 3D simulation of plasma deposition to the first wall and corresponding signal capture by the IR camera with a focus on tokamaks WEST (IRFM CEA Cadarache) and ITER. The produced synthetic signal results in a digital twin and is to be compared with experimental data. The main purpose of creating the tokamak digital twin is to better understand and control plasma power deposited to the first wall. Along with this, the digital twin allows reaching a deeper understanding of the physical processes around the inner wall during the fusion reactions, possibly leading to real-time control of the fusion reaction, as it provides a unique insight into particle movement and heated areas in the tokamak. In more detail, one of the goals of nuclear fusion reactors is to constantly monitor the temperature distribution of the inner wall using IR cameras to detect in real-time overheating parts allowing reaction before any damage occurs. Due to highly reflective materials in fusion reactors, the IR camera signal is difficult to interpret correctly since reflected light can create so-called hot spots which lead to a false interpretation of camera signals.

To better understand and interpret the IR camera signals correctly, three types of simulations are performed to arrive at the synthetic (i.e. simulated) signal on the IR camera. First, field line tracing algorithms are used to determine heat fluxes to the first wall. This is achieved with ITER/UL in-house code L2G. In field line tracing, particles are tracked along magnetic lines and intersection with solid wall is calculated. Power deposition on the intersections with the wall is then assessed based on given plasma operation parameters and passed to the thermal model which provides corresponding temperatures. The thermal model is solved with a customized solver based on the OpenFOAM kernel which uses the Finite Volume Method. Finally, the optical performance of visible light and infrared cameras is simulated, based on Monte Carlo methods using backward ray-tracing algorithms to predict the optical flow.

Corresponding temperatures on the walls are used to predict radiation and reflection from the wall in the field of view of the camera. Surface reflectivity is assessed with Cook-Torrance models, which are able to take into account experimentally verified reflectivities for first wall materials. Here the use case aims to take advantage of the ray-tracing package Raysect, developed with the specific purpose to simulate physical processes in magnetically confined nuclear devices. The goal of the simulations is to accurately predict stray light, hot spots, and uniformity. Since cameras do not only see surface temperature but also reflect light from other surfaces, coupled thermal-optics simulations under high reflectance are used to correctly describe the visible light: when reflections are subtracted, correct surface temperatures are determined. Therefore, only complete simulations (namely, digital twins) can give correct results to be seen by cameras. Finally, the proposed workflow is to be optimized and conducted with advanced high-performance computing on leading-edge HPC architectures.

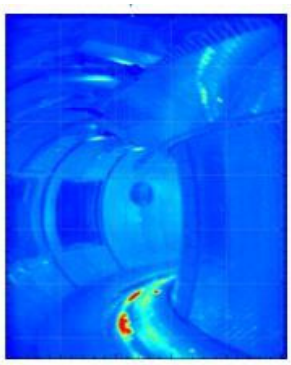


Fig. 1: Synthetic camera image of WEST tokamak during operation.

Excellerat CoE presents the main entry point for engineering stakeholders to access experts, services, and knowledge. As the engineering sector is very demanding when talking about HPC resources, EXCELLERAT partners are working on providing expertise in data management, data analytics, visualization, simulation-driven design, and co-design with Exascale computing. This will solve highly complex and costly engineering problems and create enhanced technological solutions even at the development stage.

HPC and DHInfra

Florian Atzenhofer-Baumgartner

University of Graz, Austria

The digital transformation of the Humanities has contributed to the creation of vast amounts of (multimodal) data that now often require the use of advanced methods and HPC environments to conduct meaningful and relevant research. Despite the growing use of AI applications and big data analytics within Digital Humanities (DH), there remains a considerable gap in terms of both specialized systems designed to meet the needs of DH scholars and the recognition of their research priorities in broader research agendas. These shortcomings are intensified by the latest boom of generative AI.

To address these challeges, various initiatives and working groups, such as within the EuroHPC Joint Undertaking or (national) DH organizations, have recently been established to bring the Humanities and AI closer together. These efforts aim to develop use cases that demonstrate the value of HPC in Humanities research and to reduce barriers to accessing and utilizing its resources. A major contribution to this lies in the Digital Humanities Infrastructure (DHInfra; dhinfra.at) project, which will develop and set up components for Austrian DH research in 10 institutions. It works closely together with the CLARIAH consortium, i.e., a product of the Digital Research Infrastructure for the Arts and Humanities (DARIAH) and the Common Language Resources and Technology Infrastructure (CLARIN).

The project lead – the Centre for Information Modelling at the University of Graz – puts a focus on HPC and data analytics. This work reports on current plans thereof with the output being three-fold. First, it describes potential flagship use cases that have arisen in recent DH projects and are relevant to the infrastructure being built, such as large-scale speech recognition of contemporary history video material [1], topic modelling and sentiment analysis of historical newspapers [2], or layout segmentation and object detection for ad-hoc legal document analysis [3]. Second, it proposes three different clusters and configurations to tackle the needs of DH: One classical setup for dedicated training and fine-tuning of large models (e.g., with scheduling via SLURM), another one that fosters exploration and collaboration (e.g., using a customized JupyterHub with different kernels), and lastly a system for scheduled inference (e.g., employing TorchServe or ONNX runtime engines). Third, this work critically reflects on each system's limitations that are to be anticipated in the implementation and governance stage.

As such, the project highlights the potential for HPC to improve Humanities research. Furthermore, it underscores the importance of ongoing collaboration between DH scholars and HPC infrastructure providers to ensure that the developed solutions are both effective and sustainable.

References

[1] Fleck, M., and Göderle, W., "Wav2vec and Its Current Potential to Automatic Speech Recognition in German for the Usage in Digital History: A Comparative Assessment of Available ASR-Technologies for the Use in Cultural Heritage Contexts," arXiv preprint arXiv:2303.06026, (2023).

[2] Venglarova, K., et al., "Tracing the Transformation of the Labour Market through Historical Job Advertisements," https://doi.org/10.5281/ZENODO.10698346, (2024).

[3] Vogeler, G., et al., "Building a Virtual Research Environment to Move from Digital to Distant Diplomatics (ERC Project DiDip)," https://doi.org/10.5281/ZENODO.7715522, (2023).

Croatian National Advanced Computing Resources Supek and Vrančić

Emir Imamagić, *Jurica Špoljar*, and Martin Belavić

University Computing Centre, University of Zagreb, Croatia

The University of Zagreb, University Computing Centre (SRCE) hosts two Croatian national advanced computing resources - Supek and Vrančić [1]. Deployment of these resources was one of the successfully achieved goals of the project Croatian Scientific and Educational Cloud (HR-ZOO) [2]. HR-ZOO established a national research and innovation e-infrastructure for the Croatian scientific and academic community.

Supek is a supercomputer based on HPE Cray EX2500 and HPE ClusterStor E1000 systems with sustained computing power of 1.25 PFLOPS. Supek consists of three partitions - CPU partition with 52 worker nodes providing a total of 6656 CPU cores, GPU partition with 20 worker nodes providing a total of 1280 CPU cores and 80 NVIDIA A100 and big memory partition with 2 worker nodes providing 128 CPU cores and 4 TB RAM each. All partitions are using AMD Epyc 7763 CPUs. Storage is all-flash based providing 580 TB. Finally, interconnect is based on 200 Gbit/s Slingshot technology.

Vrančić is a cloud computing platform that provides 11520 AMD Epyc 7713 CPU cores, 16 NVIDIA A100 GPUs and 57 TB of RAM. Vrančić is based on widely used open-source platforms OpenStack and Ceph. While Supek is used for classic HPC applications and heavy AI workloads, Vrančić enables running smaller applications, performing interactive computations or visualizations and establishing custom compute-demanding environments, such as Big Data or Notebook-based platforms.

Resources have been in production since April 2023 and currently are used by over 150 projects, and over 300 individual users working in 50 different institutions. In the past year support team has deployed, optimized and documented around 100 different scientific applications.

In this talk, we will present fine tuning and reviews of the batching system scheduling algorithm based on the analysis of jobs as well as feedback received from users. Furthermore, we will demonstrate our online reporting system based on R and Shiny that enables users to see various resource usage statistics. Finally, we will describe challenges with the direct liquid cooling system, native Slingshot network support in Cray MPICH and OpenMPI, as well as monitoring frameworks.

Fig. 1: Supercomputer Supek

References

[1] Dobrenić Dobriša, Imamagić Emir, Marić Ivan, Project HR-ZOO - The Final Stages of Implementation, Computational Chemistry Days (2022).

[2] Imamagić Emir, Predstavljanje usluge Napredno računanje, Srce Dani e-infrastrukture 2023.

[3] https://www.srce.unizg.hr/en/advanced-computing

Keynote Talk:

AI and HPC: Paving the Way for Europe's Digital Future

Jeanette Nilsson

Research Institutes of Sweden AB (RISE), Sweden

Europe is committed to empowering businesses and individuals to achieve a sustainable and prosperous digital future. The Digital Decade policy programme sets specific goals and targets for 2030, guiding this transformation [1]. In order to achieve the objectives of the Digital Decade, the EuroHPC JU has successfully established and enhanced a High Performance Computing (HPC) infrastructure that ranks among the most powerful in the world. The introduction of basic models such as Chat GPT in 2022 has been a game changer for Artificial Intelligence (AI) development, which has progressed at a remarkable pace. In response, the European Commission (EC) has introduced an AI innovation package, which includes the AI Factories initiative [2].

- **AI dedicated supercomputers**: The EuroHPC JU now acquires, upgrades and operates supercomputers specifically tailored for AI tasks. These machines facilitate rapid machine learning and training of large General Purpose AI (GPAI) models.
- **Broadening access**: By providing access to dedicated AI supercomputers, the EuroHPC JU aims to democratise the use of AI. Startups, SMEs and other stakeholders can use this infrastructure to explore the potential of AI.
- **Support ecosystem**: AI Factories serve as comprehensive hubs for startups and innovators. They provide resources for algorithmic development, testing, evaluation and validation of large-scale AI models. They also provide supercomputer-friendly programming facilities and other AI-enabling services.
- **Emerging AI applications**: The EuroHPC JU promotes the development of diverse AI applications based on general-purpose AI models.

To bridge the gap between AI and HPC, the Research & Innovation Advisory Group (RIAG) within the EuroHPC JU has established the AI-Data Working Group. This workgroup focuses on connecting new users to the EuroHPC infrastructure, recognising that testing and fine-tuning AI models requires different considerations than the advanced HPC community [3]. Data will be more important and new requirements for storage and security will be needed when working and developing AI models and tools. The outcome must be in line with the values of the European Digital Policy Programme, as outlined in the Ethics guidelines for trustworthy AI.

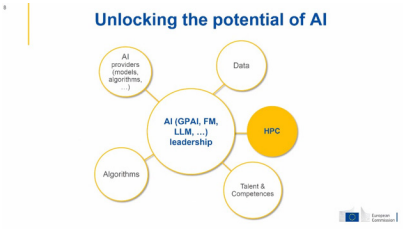


Fig. 1: AI potential for digital decade

How can we work together to create good conditions for a sustainable future for future generations?

References

[1] Digital decade policy programme, https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/europe-fit-digital-age/europes-digital-decade-digital-targets-2030_en

[2] EC AI innovation package, https://ec.europa.eu/commission/presscorner/detail/en/IP_24_383

[3] Generative AI in the European Startup Landscape 2024, https://www.appliedai-institute.de/en/hub/2024-generative-ai-study.

Live resource management in HPC applications: Resilient hybrid computing in IMS' MBMW processing pipeline

David Pfahler **and Harald Höller-Lugmayr**

IMS Nanofabrication GmbH, Brunn am Gebirge, Austria

We address the intricate challenge of managing resource limitations and hardware utilization within the Multi-Beam Mask Writer (MBMW) processing pipeline [1]. Our primary objective is to enhance stability and availability by dynamically adapting to varying computational demands during the real-time processing for mask writing with high bandwidth on an HPC cluster. It's crucial to note the inherent variability of parameters and input data and computational demand on the hardware components. In combination with an industry constantly pursuing higher complexity [2], upfront knowledge of expected resource amounts is very challenging. This uncertainty emphasizes the significance of our adaptive resource management approach in addressing the dynamic nature of the MBMW processing pipeline.

Our proposed framework integrates three key techniques. Firstly, real-time tracking of memory allocations within each process, covering both GPU and CPU utilization, provides granular insights into the specific resource requirements of individual components. This enables precise resource management. The second technique involves aggregating the tracked memory allocations based on insights from individual processes. This holistic approach offers an overview of global resource utilization, facilitating the establishment of globally tracked limitations. This strategy enhances the adaptability of resource management across the entire MBMW processing pipeline, contributing to overall system stability. With this, offloading tasks to the GPU devices are controlled by memory allocation, allowing optimal hardware utilization. The third technique focuses on the dynamic employment of down-scaling mechanisms in response to exceeded limitations. This proactive approach eliminates the risk of Out-of-Memory situations, ensuring continuous operation by adapting to resource limitations in real time. This adaptive response mechanism significantly contributes to improving the stability of the MBMW processing pipeline.

Looking forward, our approach holds the potential for further enhancement to incorporate additional resource metrics beyond CPU and GPU memory. Metrics that may prove relevant for HPC applications include CPU utilization, the number of concurrent threads and processes, the number of open file descriptors, and domain-specific metrics such as the count of input polygons for the MBMW processing pipeline.

Validation efforts encompass comprehensive experiments using real-world scenarios on an HPC cluster. The results highlight a substantial improvement in the system's ability to cater to changing computational requirements. This enhancement, demonstrated through improved stability and availability, is quantitatively measured by the system's 99 % up-time. This metric ensures a reliable assessment of the proposed approach.

References

[1] Christof Klein and Elmar Platzgummer, "MBMW-101: World's 1st high-throughput multi-beam mask writer", SPIE Photomask Technology 2016, **9985**, 998505 (2016).

[2] Kyungsup Shin, Harald Höller-Lugmayr et. al., "New multi-beam mask data preparation method for EUV high volume data", XXIX Symposium on Photomask and Next-Generation Lithography Mask Technology, Proc. SPIE, **12915**, 116-125 (2023).

Enhancing Computational Throughput in Multi-Beam Mask Writing: Leveraging Celery as a Distributed Task Queue on Heterogeneous Clusters

Paul Heistracher **and Lukas Schuller**

IMS Nanofabrication GmbH, Brunn am Gebirge, Austria

Multi-beam mask writers (MBMW) play a pivotal role in semiconductor fabrication, necessitating high-throughput and parallel processing capabilities [1, 2]. In search of a successor to our in-house dispatching framework, a potential candidate had to fulfill stringent scalability and fault-tolerance requirements to meet the industry's high standards.

This presentation delves into the practical advantages of integrating the Celery Python library [3] as a distributed task queue to optimize task dispatching on heterogeneous cluster hardware, particularly within the context of engineering the driver of a multi-beam mask writing tool. With features such as distributed task management and asynchronous communication, Celery proved to be a good choice for addressing the challenges posed by the computationally demanding task of mask writing under production workloads while simultaneously keeping the complexity manageable.

We focus on the practical benefits of leveraging Celery to decouple task submission and execution, enabling seamless distribution of computational workloads across nodes while ensuring strong scaling across expansive cluster hardware. Celery's task canvasing mechanisms are utilized to simplify the scheduling of complex workload formations while optimizing resource utilization, resulting in increased overall system efficiency.

Furthermore, Celery's flexible and extensible nature is highlighted within the specific context of creating fault-tolerant applications meeting requirements on resilience and up-time in production. The library's ability to gracefully handle task retries and rescheduling proves crucial for maintaining continuous and uninterrupted operations, aligning with the stringent requirements of semiconductor manufacturing processes.

Celery's intuitive design and straightforward implementation make it also an accessible tool for researchers seeking efficient task dispatching in distributed computing environments, enabling them to focus more on their scientific work and less on task management intricacies.

The findings of this study offer a compelling argument for the adoption of Celery as a user-friendly solution for optimizing HPC workflows, seeking simplicity and efficiency in their computational tasks both in an academic as well as an industrial context.

References

[1] Klein, C., and Platzgummer, E., SPIE Photomask Technology 2016, **9985**, 998505 (2016).

[2] Tomandl, M., Spengler, C., Klein, C., Loeschner, H., and Platzgummer, E., Journal of Micro/Nanopatterning, Materials, and Metrology **23**, 011205 (2024).

[3] Celery Documentation, Celery - Distributed Task Queue, https://docs.celeryq.dev (2024 Jan 22).

High-throughput image processing in multi-beam mask writing

Martin Jurkovič **and Mario Zauchner**

IMS Nanofabrication GmbH, Brunn am Gebirge, Austria

In 2016, IMS Nanofabrication introduced its breakthrough Multi-Beam Mask Writer (MBMW) tool for high-volume leading-edge mask production [1]. This tool addressed challenges in the evolving landscape of the mask production technology, particularly with the introduction of Inverse Lithography Technology (ILT) together with curvilinear patterning for assist features and Extreme UltraViolet (EUV) lithography for the most advanced semiconductor chip nodes. These advancements rendered single-beam mask writers impractical due to escalating writing times, making MBMWs the key technology for maintaining reasonable mask writing times [2].

The MBMW uses a 2D matrix of a quarter million tiny switchable electron beams. Controlling the beamlets is achieved by transferring pixel data to a controller in real-time. The customer design is rasterized by a heterogeneous cluster of CPU and GPU nodes and data is transferred to the controller via an infiband network. In order to maintain ca 10 hour mask write times, datarates of several hundreds of Gbit/s have to be provided. During processing of the mask, several petabytes of data are transferred to the controller, making rasterization of mask layouts a processing and memory intensive process.

Furthermore, the rasterized image data has to undergo several post-processing steps in order to obtain the desired exposure on the mask. Due to hardware limitations, these post-processing steps were implemented with an emphasis on GPU memory usage, leading to several design constraints. As MBMW customers demand smaller and smaller pixel sizes, the required datarates to maintain 10 hour mask write times continue to increase. In addition, recently introduced new use cases for our MBMW tools require even more expensive post-processing of the rasterized image. Thus, the original implementation became a major computational bottleneck and improvements were required in order to keep up with the increasing computational demands [3].

As current GPU generations offer substantially more memory, new opportunities for optimization of our post-processing algorithms became available. In this work, we present the optimization of one of our post-processing algorithms and discuss the performance gains we achieved by redesigning the implementation. In addition, we will discuss further optimizations which can be exploited under specific parametrizations of the algorithm, which are particularly relevant to the use-cases of our customers.

References

[1] Christof Klein and Elmar Platzgummer, SPIE Photomask Technology 2016, **9985**, 998505 (2016)

[2] Mark LaPedus, Semiconductor Engineering, November 2018, *Multi-Beam Mask Writing Finally Comes Of Age*, https://semiengineering.com/multi-beam-mask-writing-finally-comes-of-age/

[3] Mathias Tomandl *et al.*, "Multi-Beam Mask Writing opens up New Fields of Application, including Curvilinear Mask Pattern for High-NA EUV Lithography", accepted for publication in Journal of Micro/Nanopatterning, Materials and Metrology (JM3 2024)

Leonardo's Role in the Pre-exascale Era

Moreno Guernelli

Cineca - HPC Department, Italy

Leonardo, hosted by CINECA in Bologna Technopole, Italy, is the latest pre-exascale Tier-0 EuroHPC Joint Undertaking supercomputer. Currently, it holds the 7^{th} position in the Top500 list, exhibiting a computational performance of nearly 240 Pflops. Supplied by EVIDEN ATOS, Leonardo is powered by two newly designed compute blades, each accessible through separate SLURM partitions:

LEONARDO Booster partition
· X2135 GPU blade based on customized NVIDIA *Ampere A100-64GB* accelerators (3456 nodes).

LEONARDO Data Centric General Purpose (DCGP) partition
· X2140 CPU-only blade based on Intel *Sapphire Rapids* processors (1536 nodes).

The architecture integrates *NVIDIA Mellanox InfiniBand HDR* connectivity, which guarantees optimal performance and scalability for AI and HPC applications. The storage system comprises a *capacity* tier (∼ 100 PB) and a *fast* tier, (∼5 PB at 1.4 TB/s) offering flexibility to address demanding I/O use cases in terms of bandwidth and IOPS. Most common third-party software (such as *GROMACS, Openfoam, MATLAB, QuantumEspresso* and more) are available as *modules* grouped by functional categories (*e.g.* compilers, libraries, applications) and scientific domain. Moreover, it is possible to install additional software via *spack* environment. Access to the cluster via *ssh* protocol is secured by a two-factor authentication system.
Leonardo aims to support academic and industrial research endeavors across Europe, with resources accessible through *EuroHPC JU* or *Partners' Access Calls*.
I'll provide a breakdown of Leonardo's key features, guiding you on accessing resources, optimizing cluster efficiency, and outlining upcoming upgrades, particularly focusing on the next challenge: *LISA partition*.

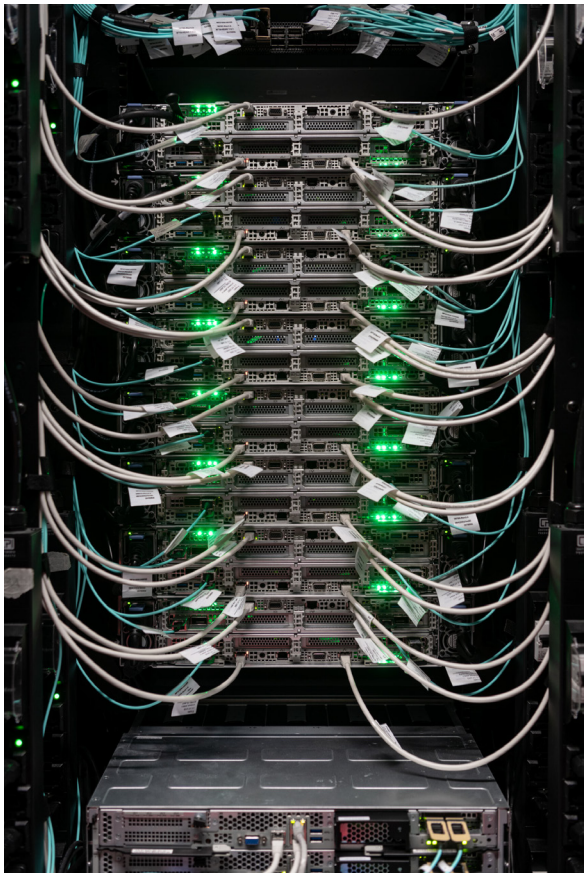


Fig. 1: Leonardo's pulsating core[1].

References

[1] https://leonardo-supercomputer.cineca.eu/

Development and supporting activities on EuroHPC Vega

Teo Prica

IZUM – Institute of Information Science, Slovenia

HPC Vega is the first launched petascale EuroHPC supercomputer, entering its fourth year of operation. It still provides the necessary infrastructure for the Slovenian scientific community, projects within the European Union through EuroHPC Calls (Benchmark, Development, Regular and Industry Access), and commercial usage of national share. Interest from various scientific domains is increasing each year, worth mentioning are artificial intelligence (AI) and machine learning (ML). Demand also comes from the industry with acquired customers hosted on the commercial share of HPC Vega. We are happy to write new success stories. That said, we'll focus on current development and supporting activities for various projects hosted on HPC Vega [1].

	EuroHPC JU	SI	Commercial	Atlas
2023-01	19.409.146	15.033.334	113.225	38.345.106
2023-02	18.546.138	19.328.112	102.780	26.136.337
2023-03	22.991.335	26.829.111	48.801	19.843.778
2023-04	18.292.839	20.749.343	7.603	22.551.656
2023-05	26.242.049	28.590.384	92.932	22.386.325
2023-06	23.981.766	31.146.674	30.557	19.578.922
2023-07	19.994.718	27.696.061	182.924	19.375.187
2023-08	13.040.568	31.573.785	100.731	26.233.564
2023-09	23.640.398	29.256.586	67.578	19.227.576
2023-10	21.336.137	27.449.299	176.017	21.892.135
2023-11	17.331.742	15.435.210	350.154	31.032.321
2023-12	17.978.835	11.244.567	5.139	42.649.684
Average	20.232.139	23.694.372	106.537	25.771.049
SUM	242.785.671	284.332.466	1.278.441	309.252.591
% of total	29,0%	33,9%	0,2%	36,9%

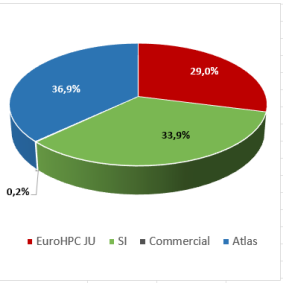


Fig. 1: HPC Vega usage by month in 2023

We are involved in a number of established projects, such as interTwin, which uses the HPC Vega infrastructure and Edge Virtual Machine (VM) for its development activities, SMASH (MCSA COFUND), where we provide on-site support for postdoc candidates in our research area (software and infrastructure for high-performance computing), provide infrastructure and offer onboarding workshops for postdocs, EUMaster4HPC for the preparation of various internships, EPICURE – EuroHPC Application Support Call. Not to forget the contributions to non-funded projects and supporting activities ATLAS, Belle 2, CASTIEL 2, we provide support for CI/CD development activities for four associated Centres of Excellence (CoE), namely MaX3, MultiXScale, ChESSE, and EXCELLENT, high-level support within LEONARDO, collaboration with EuroHPC Systems and the EuroHPC Container Forum, European Digital Infrastructure Consortium (EDIC), EuroCC 2 for the promotion with a main focus for industry, and will provide the necessary infrastructure to host national workshops where participants could improve their skills.

Due to the circumstances, we are very flexible and can react and adapt quickly to changes in the domain. We regularly strive to keep the cluster up to date, improve various aspects, including security, and incorporate innovations that we believe will have a positive impact on end users. Providing support at different levels (1st, 2nd and 3rd), including services such as Nordugrid ARC, Ceph S3 RADOS GW, dCache, CVMFS, European Environment for Scientific Software Installations (EESSI) including GPU support. Implementing two-factor authentication (2FA), updating commonly used software stacks on site, contributing to the SLING consortium and the national CernVM-FS, providing platforms for end users such as Open OnDemand, preparing a new user portal to display projects and their resource consumption, and evaluating the convergence of HPC and cloud.

References

[1] User documentation: https://doc.vega.izum.si/

Software containers in HPC

Dejan Lesjak

Jožef Stefan Institute, Slovenia

One of the challenges for modern HPC centres is providing users with straightforward and user-friendly access to HPC resources. It is important to consider the requirements of different projects, including specific software versions, hardware setups, workload isolation etc.

Maintaining environments for AI and machine learning frameworks is especially challenging in HPC centres, as they have many dependencies and are often version-dependent. However, some projects may want to use resources in multiple HPC centres and would gain from having the same environment across them.

Some projects may require legacy software, that may not run properly on different HPC sites. In such use cases, the ability to isolate the scientific software, along with associated user space is necessary. This is important for repeatability of calculations which require as similar as possible environment. Such setup can also be advantageous in industry, when the process development is complete and the application is ready for production use.

The use of HPC resources by multiple users poses security risks to both the system and other users. Containers offer an isolation aspect that can help mitigate these risks.

On Linux, which is the predominant operating system used in HPC systems, there are two general methods to provide environment isolation and portability. To ensure environment isolation and portability, containers can be used as a lighter alternative to full hardware virtualization. This avoids the added overhead to computations that comes with emulating an entire architecture.

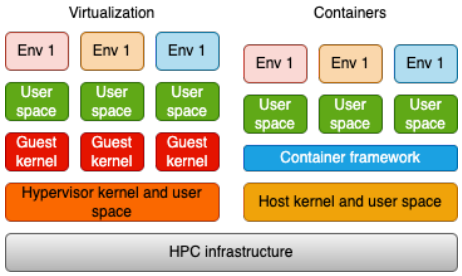


Fig. 1: Methods of isolation

This talk presents the history of using containers and their predecessors, as well as the benefits, challenges, and best practices in allowing applications to run in isolated and portable environments, independent of the host operating system and hardware as experienced at SIGNET and other SLING sites. It also explores different implementations of container frameworks such as Singularity [1] and CharlieCloud [2].

While isolating software from underlying host OS, scientific software of course also requires access to site-specific accelerators such as Nvidia GPUs and MPI interconnects like InfiniBand. However, due to isolation from the host OS, accessing such hardware presents a challenge with drivers for said hardware. The talk explores established techniques for accessing such hardware.

References

[1] Kurtzer, G. M., Sochat, V., and Bauer, M. W., Singularity: Scientific containers for mobility of compute, PloS one, 12(5) (2017)

[2] Priedhorsky, R. and Randles, T., Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC '17), Article 36 (2017)

Overcoming challenges in building scientific software for HPC

Alja Prah **and Barbara Krašovec**

Josef Stefan Institute, Slovenia

HPC plays a pivotal role in advancing scientific research across various domains, empowering researchers to solve complex problems at unprecedented scales. However, building and maintaining scientific software on heterogeneous and understaffed HPC systems poses significant challenges that hinder the efficient utilization of these powerful resources. Most scientific software relies on externally developed dependencies, resulting in a myriad of requirements for each distinct build. Resolving these dependencies consistently across various systems poses a challenge, commonly known as 'dependency hell,' impacting both developers and end users of application codes. Additionally, the European Commission is shifting its focus towards integrating the broader Artificial Intelligence (AI) community into the EuroHPC ecosystem [1], which poses novel challenges since AI and HPC require different and highly specific sets of skills and expertise. This abstract sheds light on the multifaceted problems associated with scientific software development on HPC platforms and explores potential solutions, emphasizing initiatives such as the Epicure project, EuroHPC Centers of Excellence (CoEs), the utilization of containers and the European Environment for Scientific Software Installations.

Epicure [2], a novel European initiative, aims to establish and operate a distributed but coordinated HPC application support service. This initiative focuses on providing comprehensive support to encourage optimal utilization of HPC systems by scientists and researchers across Europe. EuroHPC Centers of Excellence [3] (i.e. MaX, MultiXscale, ChEESE-2P and more) play a crucial role in addressing challenges related to scientific software development in the exascale era. By fostering collaboration between domain experts and HPC specialists, CoEs facilitate the creation of optimized software tailored to unique requirements of diverse research domains.

Containerization (championed by the EuroHPC container forum) is another possible solution to the challenges associated with HPC software development. By encapsulating applications and their dependencies in lightweight, portable containers, researchers can ensure reproducibility, scalability, and ease of deployment across different HPC environments. Containers also simplify the process of managing software dependencies, mitigating compatibility issues and streamlining the deployment pipeline. The European Environment for Scientific Software Installations (EESSI) represents a collaborative effort within the European HPC community to standardize the environment for scientific software installations. By offering a standardized framework EESSI simplifies the deployment of scientific applications on diverse HPC architectures.

In conclusion, this conference abstract introduces some of the approaches addressing challenges in building scientific software on HPC platforms. Through collaborative efforts exemplified by EuroHPC Centers of Excellence, containerization, and initiatives like Epicure and EESSI, the HPC community can pave the way for a more accessible, interoperable, and efficient scientific computing ecosystem, ultimately benefiting scientists and researchers across Europe.

References

[1] European Commission. (16 Nov 2023). Commission opens access to EU supercomputers to speed up AI development.

[2] EuroHPC JU. (13 Feb 2023). New call for developing a EuroHPC application support service.

[3] EuroHPC JU. (26 Jan 2023). Kick-off of 10 Centres of Excellence in HPC to support the transition towards exascale.

Lessons learned/experience from the battlefield: HPC user support

Violeta Šikaleska* and *Samo Lorenčič

IZUM - Institute of Information Science, Slovenia

User support is a vital aspect of the HPC Vega system, which offers a variety of hardware resources and software for diverse computational needs. In this presentation, we will share the lessons learned from engaging with user queries and tickets and provide solutions for common technical challenges.

We will provide guidance for users on how to use the allocated resources, handle multiple project access, and establish resource usage reports effectively. We will also discuss the optimization of required resources in scripts, such as memory and CPU allocation. Understanding the differences between the largemem partition and the CPU partition is crucial for efficient resource management. Furthermore, we will introduce the software modules, libraries, and containers available on Vega, and offer tips for using software, optimising scripts, and improving performance on the cluster. We will explain the different file systems and their suitability for different workloads.

	SUM	SI	SI [%]	EuroHPC JU	EuroHPC JU [%]
2023-01	99	68	68,7%	31	31,3%
2023-02	69	43	62,3%	26	37,7%
2023-03	107	68	63,6%	39	36,4%
2023-04	73	45	61,6%	28	38,4%
2023-05	72	34	47,2%	38	52,8%
2023-06	77	37	48,1%	40	51,9%
2023-07	78	45	57,7%	33	42,3%
2023-08	71	42	59,2%	29	40,8%
2023-09	80	49	61,3%	31	38,8%
2023-10	88	64	72,7%	24	27,3%
2023-11	64	42	65,6%	22	34,4%
2023-12	57	34	59,6%	23	40,4%
SUM 2023	935	571	61,1%	364	38,9%
Average (per month)	78	48	61,1%	30	38,9%

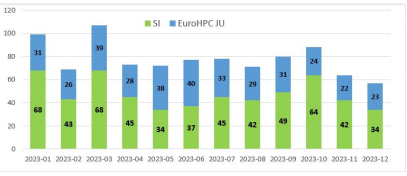


Fig. 1: Number of tickets per month in 2023

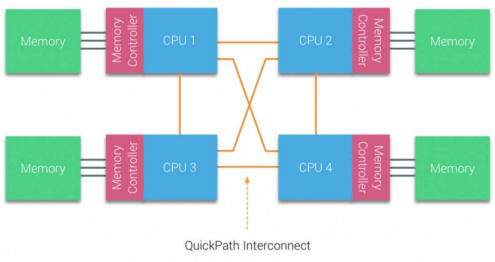


Fig. 2: NUMA memory design.

In addition, we will compare multithreading with utilising physical cores and highlight the importance of CPU pinning. When executing multicore/multinode jobs, users often face performance issues caused by either communication/network or memory overhead. In the NUMA memory design, the memory access time depends on the memory location relative to the processor. To achieve optimal performance, the ability to establish a mapping between virtual CPU and physical core is required, which is covered by efficient CPU pinning. Inefficient task binding in MPI jobs can lead to poor performance. There are strategies to avoid such issues. Lastly, we will address network congestions that may arise when dealing with large jobs and potential bottlenecks they may cause.

With these insights in mind, users can unleash the full potential of the HPC Vega system, boosting their performance and efficiency. The HPC Vega system is a cutting-edge platform that empowers users to tackle complex and diverse computational challenges.

References

[1] User documentation: https://doc.vega.izum.si/

[2] NUMA memory design: https://frankdenneman.nl/2016/07/07/numa-deep-dive-part-1-uma-numa/

Poster:

European Master for High Performance Computing (EUMaster4HPC) project

Tomas Kozubek

IT4Innovations, VSB – Technical University of Ostrava, Ostrava-Poruba, Czech Republic

Launched in January 2022 with a substantial budget of € 7 million, the EUMaster4HPC project is driving progress in High-Performance Computing (HPC) education across Europe. Funded by the EuroHPC Joint Undertaking, the project plays a key role in harmonising HPC education by developing a common European Master's programme [1]. The program has successfully developed and co-designed a comprehensive 120 ECTS curriculum tailored to meet industry needs and offering four key specialisations: Numerical and Data Specialist for Science Domain, Performance Analyst and Advisor, System Developer and Support, and System Architect. These areas are complemented by transversal skills that are essential in the modern computing landscape.

To enhance global accessibility and foster international cooperation, EUMaster4HPC has established over 30 dual-degree agreements, promoting an enriched educational experience through diverse academic environments. The EUMaster4HPC program has attracted significant interest, bringing in a broad range of talented applicants from 44 different countries. **Now, the call for applications for the third cohort is open until July 15, 2024, with new students set to begin their studies in the autumn semester of 2024/2025.**

Supporting mobility and inclusivity, the project has awarded 62 students across the first two cohorts with mobility grants and tuition waivers, and it is poised to continue this support for a forthcoming third cohort. The initiative's commitment to practical learning is evidenced by the organisation of 30 internships with partners in industry and academia, providing students with real-world applications.

The Moodle platform developed by the project is a cornerstone for HPC education, offering a variety of teaching resources and newly developed MOOCs. These online courses cover important topics such as Massive Parallel Programming on GPUs, Introduction to Quantum Computing, and Energy Aware Parallel Computing, which are crucial for students aiming to excel in the HPC field. In addition to these educational advances, EUMaster4HPC has facilitated important academic activities, including workshops, challenges, and summer schools. **The next EUMaster4HPC summer school is planned in August 2024 at the IT4Innovations National Supercomputing Centre [1]. This event will focus on HPC in Data Science and aims to train up to 50 students.**

The future of the EUMaster4HPC project appears bright, with plans to expand its network of partners and awarding universities. The program is committed to continually updating its curriculum to align with the changing needs of the industry and to solidify its position as a cornerstone of European HPC education.

Acknowledgement: The project has received funding from the European High-Performance Computing Joint Undertaking under grant agreement No. 101051997.

References

[1] https://eumaster4hpc.eu

Poster:

MaX Centre of Excellence and the EuroHPC ecosystem (after the 1st year of MaX3)

***Jan Jona Javoršek*, Alja Prah, and Florian Klauser**

Jožef Stefan Institute, Slovenia

MaX, Materials design at the Exascale, has been established in 2016 as a H2020-funded Centre of Excellence with the mission to enable materials modeling, simulations, discovery and design at the frontiers of HPC (High-Performance Computing), HTC (High-Throughput Computing), and data analysis technologies. As a European pilot in open data, the centre invested in user community support and software maturity [1], but its central challenge lies in a redesign of quantum material simulation flagship software, data and workflow ecosystems [2] to enable a paradigm shift as computational design and discovery packages would become able to harness the rising computational power to identify novel materials with improved properties and performance [3].

As MaX became MaX3, an EuroHPC centre of excellence, its scope expanded to the new, wider eco-system and towards exascale. We describe how MaX3 uses schools, workshops, benchmarking campaigns and CI/CD challenges to expand the efforts in software maturity, community support, code availability and platform support to meet workflow, data management, and scalability challenges of the project's users.

Requirements of optimization, testing and deployment to EuroHPC sites in order to provide flagship software for materials science [4] to the user communities dictate the shape of this effort.

We will demonstrate the work of the MaX3 Center of Excellence with an overview of how we are establishing CI/CD infrastructure to support developers with security, quality, scalability, and continuous deployment tools that will help extend our efforts throughout the EuroHPC ecosystem.

We conclude the presentation with an overview of some of the technical challenges of the implementation of automated CI/CD with suitable security and consistency provisions in the context of current HPC environments to provide additional information for interested parties.

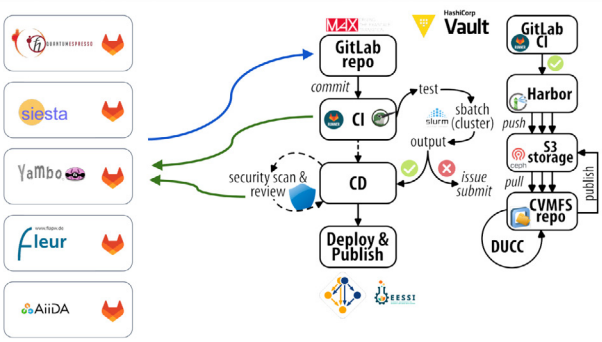


Fig. 1: CI/CD pipelines for integrity, security and deployment towards HPC.

References

[1] Ordejón, P., et al., *Final report on specific community needs and challenges; MAX-1 Deliverable 2.2* (2018). https://drive.google.com/open?id=1x2REqFayGQk-upKUte6LZH7MKtlZ46v5

[2] J. Chem. Phys., Special issue on Electronic Structure Software (2020). https://pubs.aip.org/jcp/collection/1140/Electronic-Structure-Software

[3] Goldbeck, G., and Court, C., The Economic Impact of Materials Modelling: Indicators, Metrics and Industry Survey. Goldbeck Consulting Ltd., Cambridge, UK (2016). https://zenodo.org/record/44780

[4] Carnimeo, I., et al., QUANTUM ESPRESSO: One Further Step toward the Exascale, J. Chem. Theory Comput. (2023). https://doi.org/10.1021/acs.jctc.3c00249

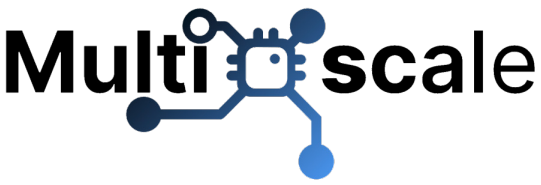
Poster:

MultiXscale – a EuroHPC JU Center of Excellence

Matej Praprotnik

Laboratory for Molecular Modeling, National Institute of Chemistry, Ljubljana, Slovenia

In the rapidly evolving landscape of scientific research, the advent of exascale computing has unlocked unprecedented possibilities. Exascale simulation presents immense scientific potential, promising groundbreaking insights across diverse domains. However, realizing this potential presents formidable challenges, encompassing both the intricate hardware ecosystem and the development of applications capable of effectively harnessing exascale capabilities. Among these challenges, multiscale modelling emerges as a particularly complex endeavor, requiring the seamless integration of methods, applications, and libraries across different scales. MultiXscale Center of Excellence (CoE) [1] addresses the inherent challenges of multiscale modelling by providing a suite of complementary tools to facilitate the creation and adoption of cutting-edge applications and workflows. The goal of the MultiXscale CoE is a software ecosystem for the efficient implementation of multiscale simulations on supercomputers, especially developing software and workflows to solve the pressing challenges of modern society in connection with energy, biomedicine and civil transport. Open source software will be accessible to researchers and engineers and will be easy to access and to use, with guaranteed support across multiple architectures. MultiXscale CoE aims to enhance performance, productivity, and portability in multiscale simulations, leveraging the scientific expertise of the CECAM network [2] and the technical strength of the European Environment for Scientific Software Installation (EESSI) collaboration [3]. In the first year, MultiXscale CoE achieved significant milestones in automating performance and scalability measurements of key software packages, employing a testing framework based on ReFrame. This testing suite is operational on EuroHPC systems Karolina and Vega, facilitating collaboration with CASTIEL 2 in Continuous Integration (CI). Efforts were also directed towards establishing a central stack of scientific software and dependencies, which also serves as the foundation for a CI infrastructure to expedite application development. The shared software stack deployed on EuroHPC JU systems Vega and Karolina encompasses packages such as ALL, ESPResSo, LAMMPS, and OpenFOAM, with plans to incorporate additional software like waLBerla. The initiative seeks to extend this environment beyond EuroHPC resources, making it accessible across various levels of infrastructure, including platforms like the European Open Science Cloud. In conclusion, MultiXscale represents a concerted effort to advance multiscale modelling and exascale computing for scientific discovery. By addressing technical challenges and fostering collaboration across scientific and technical domains, MultiXscale strives to unlock the full potential of exascale computing in driving transformative scientific breakthroughs.



References

[1] https://www.multixscale.eu

[2] https://www.cecam.org

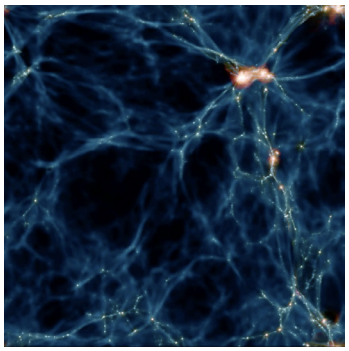
[3] https://www.eessi-hpc.org

Poster:

SPACE – Scalable Parallel Astrophysical Codes for Exascale Center of Excelence

Lubomir Riha

IT4Innovations, VŠB - Technical University of Ostrava, Czech Republic



In A&C today, HPC-based numerical simulations are outstanding instruments for scientific discovery. They represent essential tools and theoretical laboratories able to investigate, interpret and understand the physical processes behind the observed sky. For these laboratories, the efficient and effective exploitation of exascale computing capabilities is essential. Exascale systems, however, are expected to have a heterogeneous unprecedented architectural complexity, with a significant impact on simulation codes. Consequently, the SPACE CoE is aiming at extensively re-engineer seven target codes (Open GADGET, CHANGA/GASOLINE, PLUTO, iPic3D, RAMSES, FIL, and BHAC) to engage with new computational solutions and adopt innovative programming paradigms, software solutions, and libraries.

In addition, SPACE is addressing the high-performance data analysis of the data torrent produced by exascale A&C simulation applications, also with machine-learning and visualization tools. The deployment of applications running on different platforms is being facilitated by federating capabilities focusing on code repositories and data sharing, and integrating European astrophysical communities around exascale computing by adopting software and data standards and interoperability protocols.

Therefore, SPACE CoE is promoting and supporting cutting-edge science through effective exploitation of exascale resources, by anticipating the challenges of the exascale era, adopting standardized and interoperable solutions for software in accordance with FAIR principles and promoting the related know-how within the community and beyond. This allows to extend the user base and enable an increasing number of researchers to effectively use and adopt advanced HPC solutions. During the first year of its lifespan, SPACE managed to address several actions toward achieving its scientific objectives, identifying 20 scientific cases, completing a first profiling and benchmarking activity as well as porting all the codes on IT4I clusters; managed to identify the use cases for visualization and suitable tasks to exploit machine learning; managed to broadly disseminate, communicate, as well as shape effectively the preliminary exploitation of the seven lighthouse exascale A&C applications; managed to establish fruitful collaborations, innovation activities, training initiatives, and undertake synergies and collaboration with complementary grants and CASTIEL2 including user-engagement, thus strengthening and securing the overall project's impact. Most of these results are presented in public deliverables of the project [2].

SPACE CoE is funded by the European Union. It has received funding from the European High Performance Computing Joint Undertaking and from Belgium, the Czech Republic, France, Germany, Greece, Italy, Norway, and Spain under grant agreement No. 101093441.

References

[1] SPACE CoE project website: www.space-coe.eu, (2024).

[2] SPACE CoE project public deliverables: https://www.space-coe.eu/publications.php, (2024).

Poster:

EXCELLERAT CoE: Tackling Next-Generation Engineering Challenges in the Exascale Era

Sophia Honisch[a], *Tina Črnigoj Marc*[b], and the EXCELLERT P2 Consortium

[a] *High-Performance Computing Center Stuttgart, Germany*
[b] *Arctur d.o.o., Slovenia*

As the successor of the EXCELLERAT Centre of Excellence for Engineering Applications, EXCELLERAT P2 continues the fruitful partnership of Europe's leading HPC centres, application specialists, and supporting partners who have worked with and offered their expertise and knowledge to engineering companies and researchers from across industry and academia for the past three decades. EXCELLERAT is a single access point for expertise on how data management, data analytics, visualisation, simulation-driven design and co-design with high-performance computing (HPC) can benefit engineering.

Application codes are the core of the EXCELLERAT P2 project since they allow for achieving cutting-edge results of engineering objectives. The codes EXCELLERAT focuses on in project P2 are CODA, AVBP, m-AIA, Alya, Neko, FLEW, OpenFOAM and Elmer. EXCELLERAT P2 has based its evolution and developments on six chosen use cases, requiring the full range of HPC usage profiles, from small-scale capacity type applications, all the way to hero run type large-scale applications.

The advancements of the use cases and further applications beyond the current state of the art, as well as the results, which will be derived by the execution of the use cases and the further applications, will have a clear impact in Europe. The holistic view of EXCELLERAT P2 on the usage of HPC, High-Performance Data Analytics (HPDA) and Artificial Intelligence (AI) is needed, if engineering applications shall be supported to increase European competitiveness. Thus, the developments done in EXCELLERAT P2 will increase the European competitiveness in HPC applications, hardware design and HPC service provisioning to a new level [1].

To properly support the engineering community, EXCELLERAT Services for engineers and industrial end-users, developers, technology providers, academics and researchers who are working in engineering sectors, like manufacturing, automotive, energy, aerospace and climate are offered on the EXCELLERAT Service Portal [2].

Fig. 1: CODA is a Computational Fluid Dynamics (CFD) solver for the simulation of aircraft aerodynamics and was developed in a joint effort between the German Aerospace Center (DLR), the French Aerospace Lab (ONERA), and Airbus. Copyright: DLR (CC-BY 3.0) [3].

EXCELLERAT P2

References

[1] EXCELLERAT P2: https://www.excellerat.eu

[2] EXCELLERAT Service Portal: https://services.excellerat.eu

[3] EXCELLERAT P2 - Success Story: https://www.excellerat.eu/success-story-transparent-integration-of-emerging-hpc-technologies-into-the-computational-fluid-dynamics-software-coda

Poster:

FF4EuroHPC: SME Innovation Through HPC

Tina Črnigoj Marc[a] **and the FF4EuroHPC Consortium**

[a] *Arctur d.o.o., Slovenia*

The FF4EuroHPC project [1] ran between September 2020 and October 2023 as the third phase of the well-known Fortissimo projects. The project was funded by the European Commission under the H2020 EU Framework Programme for Research and Innovation with a total budget of €9.9 million. The project aimed to facilitate access to High-Performance Computing-related technologies for European Small & Medium Enterprises (SMEs) and thus increase the innovation potential of the European industry. Whether the SME was running high-resolution simulations, conducting large-scale data analyses, or incorporating AI applications into its business or service workflows, FF4EuroHPC assisted SMEs in connecting their businesses with cutting-edge technologies. Two open calls targeting the highest quality experiments involving innovative, agile SMEs were offered through the project. The focus was on key European industrial sectors, particularly SMEs from the manufacturing and engineering sectors. The partners identified and addressed specific business challenges for the SMEs, linked to particular industrial sectors or, in some cases, multiple industrial sectors. By implementing HPC, Artificial Intelligence (AI), Machine Learning (ML), High-Performance Data Analytics (HPDA), and other state-of-the-art technologies, the partners were able to develop unique products, innovative business opportunities, and become more competitive.

42 experiments met the open call requirements, successfully passed the evaluation process, and were selected for funding. All experiments were successfully concluded, generating a success story that highlights the expected business benefits for the participating SMEs. The success story also presents the potential impact of the experiment's results in economic terms, as well as societal or environmental challenges. The Fortissimo project continues its mission since the fourth project phase, FFplus, is expected to commence in the first half of 2024.

Acknowledgements: *FF4EuroHPC project has received funding from the European High-Performance Computing Joint Undertaking (JU) under grant agreement No 951745.*

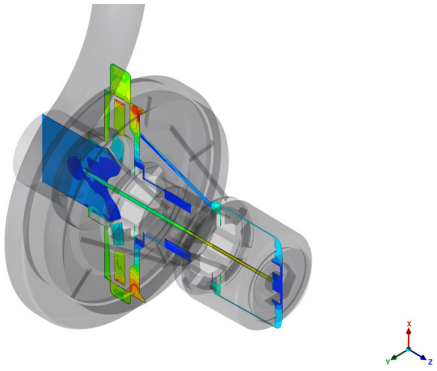


Fig. 1: This experiment investigated and optimized magnetic-drive centrifugal pumps using Cloud-based HPC to improve performance and develop new products. Copyrights: FF4EuroHPC [2].

References

[1] FF4EuroHPC: https://www.ff4eurohpc.eu

[2] FF4EuroHPC – Success Stories: https://www.ff4eurohpc.eu/en/success-stories

Poster:

NCC Czechia: HPC World Guide

***Tomas Karasek*, Karina Pesatova, and Katerina Beranova**

IT4Innovations, VSB – Technical University of Ostrava, Czech Republic

The National Competence Centre for HPC in the Czech Republic (NCC Czechia) was established in 2020 as one of the 31 NCCs across Europe under the auspices of the EuroCC project. Since 2023, its activities have continued under the EuroCC 2 project.

NCC Czechia provides services, access to the knowledge and support for the use of High-Performance Computing (HPC) and associated technologies such as High-Performance Data Analytics (HPDA), Artificial Intelligence (AI), and Quantum Computing (QC) for all stakeholders from the academia, industry, and public institutions not only in the Czech Republic but also Europewide.

We aim to increase the uptake of HPC technologies by stakeholders in the Czech Republic, increase their awareness and preparedness, and improve their digital skills related to HPC and associated technologies. In this poster, activities of the NCC Czechia, consisting of dissemination and communication activities [1], training activities [2], workshop organisation, as well as collaboration with the industry, academia and public sector, and collaboration with other NCCs will be presented. The poster will present highlights of those activities with a focus on the success stories of collaboration with Small and Medium Enterprises (SMEs). The project's objective with Drazni Revize s.r.o was to perform a proof-of-concept of the new concept of railroad scales. The concept proposed by the NCC Czechia leverages numerical modelling and simulation and HPC to prepare necessary synthetic data sets for the training of Convolutional Neural Networks (CNN). After the CNN training, this concept was explored even further, and the use of Physics-informed Neural Networks (PINNs) was studied.

The main goal of our cooperation with INFER WAY s.r.o. presented on the poster was to develop and then verify suitable procedures for processing input architectural data that would enable the generation of visually attractive outputs. Primarily, it was about methods to appropriately modify and augment the 3D scene to generate photorealistic quality video footage. Furthermore, the aim was to design procedures and modifications to the scene to allow interactive scene exploration in VR while incorporating maximum visual quality. In both cases, a path trace rendering was used. This allowed us to achieve high visual quality. In the case of generated video footage, it was used to render out all the frames.

References

[1] https://www.eurocc-czechia.cz/en/home/

[2] https://www.research.net/r/EuroCC_training_survey

Poster:

Coopetition as a model of cooperation in the Polish National Competence Center

Beata Krawczyk-Bryłka

Gdańsk University of Technology, Poland

Coopetition is a strategy of cooperation between competing entities, with the primary goal of joint creation of value. The positive effects of coopetition may concern costs, risks, economies of scale and research and development activities, as well as access to external knowledge and resources. It supports cooperating organizations in coping with VUCA (volatility, uncertainty, complexity, ambiguity) world challenge, which benefits innovation and market performance. Coopetition favours increasing competitive advantage because it allows to create products and services partners could not develop independently. Coopetition often concerns organizations that operate in the field of technological innovation or implement complex R&D projects [1].

The Polish National Competence Center operates based on five independent supercomputing centres. The centres cooperate with each other as part of the EuroCC project and create the Polish NCC, supporting each other in achieving the project's goals.

Cooperation between Polish supercomputer centres meets three primary conditions for coopetition [2]:

- mutual dependence – alliance partners need each other,
- mutual interests – alliance partners have a common interest,
- mutual benefits – all parties receive what they need in coopetition.

The cooperation undertaken within the NCC results in increased competitiveness in the Polish market due to the following:

- joint marketing activities, resulting in building awareness of HPC resources in Poland,
- facilitating reaching business and scientific clients due to integrated activities in the promotion of the portfolio of implemented projects,
- the variety of services thanks to the diversified employees' competencies and integration of HPC resources,
- joint organization of training – supra-regional promotion and reaching customers associated with individual centres,
- creating a map of HPC services, offering services tailored to the needs of local customers while using the social and technological capital of collaborating centres.

The combination of cooperation and competition also carries certain risks, such as opportunistic behaviour. However, the coopetition model works well in the case of the Polish National Competence Center due to:

- the mutual reputation of partners based on high employee competencies,
- good relationships resulting from experience in implementing projects together,
- framework of EuroCC project, where tasks are divided between individual centres.

References

[1] Yoon, S.Y., Jee, S.J. & Sohn, S.Y. Mapping and identifying technological coopetition: a multi-level approach. Scientometrics 126, 5797–5817 (2021). https://doi.org/10.1007/s11192-021-04005-x

[2] Kreft, J., & Leja, K. (2020). Koopetycja w trzech odsłonach. (Coopetition in three scenes). 1-160

Poster:

The Role of Public High Performance Computing Infrastructures in Advancing AI Across European Industries and Academia

Endri Deliu[a,b], ***Eva Gergely***[a,b], **Thomas Mayerhofer**[a,c], **and Markus Stöhr**[a,d,e]

[a] *EuroCC Austria*
[b] *University of Vienna, Austria*
[c] *INiTS/ Vienna's High-Tech Incubator, Austria*
[d] *BOKU University, Austria*
[e] *Advanced Computing Austria GmbH, Austria*

With the emergence of high-cost AI foundational models like GPT-4 and Google Gemini Ultra [1], HPC systems, especially publicly funded supercomputers, can evolve into valuable infrastructures for building large language or multimodal models for industry and academia.

This poster highlights challenges and solutions in the broader use of public HPC infrastructures for AI, showcasing how publicly funded institutions like **EuroCC Austria** can assist in resource allocation, cost estimation, expert consultation and more to enable effective utilization of supercomputing resources.

Additionally, we propose a **multi-tenant AI platform** atop HPC systems, addressing the need for consistent tracking and management of AI inputs and outputs, including configuration, models, data, and metrics [2][3]. This platform also facilitates data and model lineage, debugging, job and artifact isolation, security, orchestration of arbitrary compute steps and result publication steps within HPC systems. We show how a single unified programmatic interface for job lifecycle can be utilized to abstract multiple heterogeneous HPC clusters.

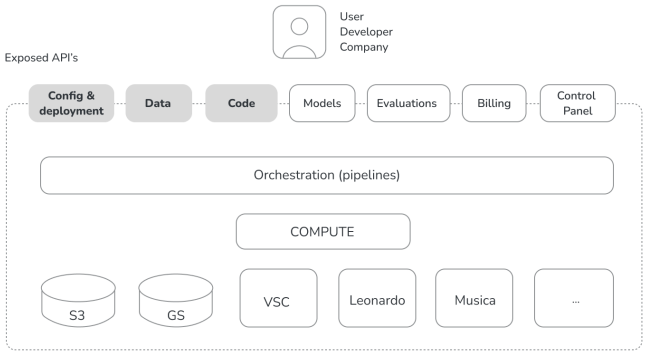


Fig. 1: AI (API) Platform for HPC Clusters

As industrial and academic needs evolve, the importance of **reliability, repeatability, reproducibility, and observability in AI workloads** becomes clear [3]. Proactive measures by HPC providers are vital to meet these evolving demands. EuroCC Austria's dedication to AI integration through consulting and strategy development promotes innovation and adoption in European industries and academia.

References

[1] AI Index Report 2024. https://aiindex.stanford.edu/report/

[2] Kreuzberger, D., Kühl, N., & Hirschl, S., Machine Learning Operations (MLOps): Overview, Definition, and Architecture (2022). https://doi.org/10.48550/arXiv.2205.02302

[3] Bosch, J., Olsson, H.H., & Crnkovic, I., Engineering AI Systems: A Research Agenda (2020). https://doi.org/10.48550/arXiv.2001.07522

Poster:

Band structure formation in metal-organic nanostructures

Dominik Brandstetter[a], **Daniel Baranowski**[b], **Andreas Windischbacher**[a], **Iulia Cojocariu**[b], **Simone Mearini**[b], **Vitaliy Feyer**[b,c], **Claus Michael Schneider**[b,c,d], **and Peter Puschnig**[a]

[a] *Institute of Physics, University of Graz, Austria*
[b] *Peter Grünberg Institute (PGI-6), Jülich Research Centre, Germany*
[c] *Faculty of Physics and Center for Nanointegration Duisburg-Essen, University of Duisburg-Essen, Germany*
[d] *Department of Physics and Astronomy, UC Davis, USA*

Ab-initio density functional theory calculations for heterotypic systems consisting of organic molecules and transition metals pose a sizeable challenge even for modern supercomputer clusters using highly optimized libraries with low-level implementation. This is because they combine two arguably opposite types of electronic behaviors which often require an advanced treatment of the exchange-correlation energy to accurately capture all relevant phenomena.

In this contribution, we present simulations for two-dimensional extended layers of a transition metal complex on a metal surface. Specifically, we study the growth of a well-ordered monolayer of 1,2,4,5-tetracyanobenzene (TCNB) and Ni on an Ag(100) surface (Fig.1). Experimentally an energy-dispersive electronic band structure has been observed indicating the formation of a metal-organic framework (2D-MOF), as has been confirmed by LEED and STM/STS measurements.

To gain deeper insight into the nature of these emerging frameworks, we have performed density functional theory calculations employing the Vienna Ab-initio Simulation Package (VASP) modeling the different phases of the 2D-MOF using a repeated-slab approach. The geometry optimization of more than a hundred atoms per unit cell with a converged set of parameters was only possible with the use of heavy parallelization on the Vienna Scientific Cluster 5 (VSC5). The nature of transition-metal complexes required allowing for a spin-polarization of the system as well as Hubbard-U corrections to the exchange-correlation functional, further increasing the computational cost.

Our calculations demonstrate the formation of hybrid states between the Ni d-states and the frontier orbitals of TCNB resulting in a dispersive band structure formation both in the freestanding layer as well as adsorbed on the Ag(100) surface. Analysis of the corresponding partial charge density reveals the delocalization of a π-like state across the entire monolayer.

To get further insight into the nature of this hybrid state, we have simulated the photoemission angular distribution. To this end, we approximate the photoemission as a one-step process and the final state of the emitted electrons as plane-waves, a technique known as photoemission orbital tomography (POT) [1]. The resulting photoemission intensity maps originating from the hybrid state are in good agreement with experiments and provide confirmation for the formation of a non-flat energy dispersion relation indicating a delocalization of hybrid Ni/TCNB states across the monolayer.

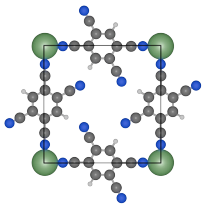


Fig. 1: Unit cell of $Ni(TCNB)_2$ on Ag(100).

References

[1] Puschnig, P. and Ramsey, M.G., Encyclopedia of Interfacial Chemistry **326**, 380 (2018).

Poster:

Quantum Vacuum Simulations in the Heisenberg–Euler Weak-Field Expansion

Andreas Lindner[a,b], **Baris Ölmez**[a], **and Hartmut Ruhl**[a]

[a] *Arnold Sommerfeld Center for Theoretical Physics, LMU Munich, Germany*
[b] *Advanced Computing Austria GmbH, Austria*

The Heisenberg–Euler theory of the quantum vacuum supplements Maxwell's theory of electromagnetism with nonlinear photon–photon interactions [1]. These originate in vacuum fluctuations, a key prediction of quantum theory. Due to vast progress in the field of laser technology in recent years, vacuum polarization can be triggered in the lab by colliding high-intensity laser pulses, leading to a variety of interesting novel phenomena. Since analytical methods for highly nonlinear problems are generally limited and since the experimental requirements for the detection of signals from the nonlinear quantum vacuum are high, the need for numerical support is apparent.

A highly accurate numerical scheme for solving the nonlinear equations due to the leading orders of the Heisenberg–Euler weak-field expansion is presented. The algorithm possesses an almost linear vacuum dispersion relation even for comparably small wavelengths and incorporates a nonphysical modes filter [2]. The solver is implemented in C++ and has been tested in one spatial dimension against a set of known analytical results for vacuum birefringence and harmonic generation [3]. More complex scenarios quickly become extremely costly with respect to computing resources and thus require the use of HPC systems. While a prominent feature of the numerical scheme is that its order is arbitrary and high orders render accurate results on comparably small simulation lattices, for distributed computing the order also determines the MPI communication load. With improved scalability, however, simulations in three spatial dimensions, capturing all physical dispersion effects, have become conductible over the course of the last two years. Selected results are demonstrated.

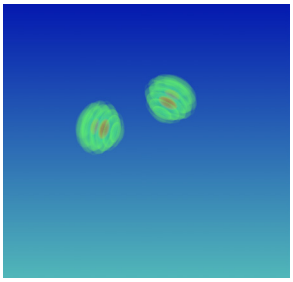


Fig. 1: Colliding laser pulses.

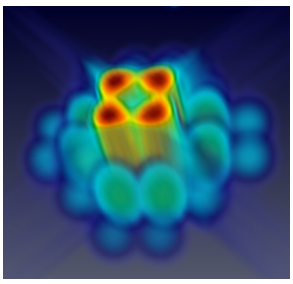


Fig. 2: Nonlinearly generated harmonics arise in frequency space.

References

[1] Heisenberg, W. and Euler, H., Folgerungen aus der Diracschen Theorie des Positrons, Z. Phys., **98**, 714 (1936).

[2] Lindner, A., Ölmez, B., and Ruhl, H., Numerical simulations of the nonlinear quantum vacuum in the Heisenberg–Euler weak-field expansion, J. Comput. Phys. X **17**, 100124 (2023).

[3] Lindner, A., Ölmez, B., and Ruhl, H., HEWES: Heisenberg–Euler weak-field expansion simulator, Software Impacts **15**, 100481 (2023).

Poster:

Deep learning for scientific data compression

Michal Kravčenko **and Tomáš Brzobohatý**

IT4Innovations, VŠB - Technical University of Ostrava, Czech Republic

The increasing power of the available supercomputing infrastructure and forthcoming exascale performance enables rapidly more accurate physical simulations, such as detailed high-fidelity CFD simulations, atmospheric Kelvin-Helmholtz instabilities, detailed hydrogen combustion processes, etc. However, as the accuracy of simulations increases, the requirements for storage space also increase quickly. Results from large-scale transient CFD simulations are typically on the order of tens of terabytes to units of petabytes. Supercomputing systems generally increase computational power much faster than the increase in storage space. The standard user space quota is in the order of tens of terabytes. An important issue is the possibility of compressing extensive 3D volumetric data so that, in some cases, simulation results can be stored, transferred and visualised even on infrastructures with limited storage and memory capacity.

We present a modified version of Fourier mapping functions [1] for learning the mapping from 3D coordinates space (x,y,z) to an output space containing, in our use cases, simulation variables. We use dynamic Fourier features mapping to (i) decrease the input size in the learning process and (ii) enable dynamic learning of the optimal distribution of features mapping. Several use cases from different types of simulations are presented for both structured and unstructured meshes. For example, Fig. 1 shows a cut of 3D volumetric data compression of combustion simulation on an unstructured 50 Million cells mesh containing 14 scalar variables in each cell. The neural network can compress the simulation data from 2.3 GB to 25 MB for this single time step.

Due to the input size, a distributed data-parallel approach for training on multiple GPU accelerators was used. An essential feature of the neural network with Fourier features mapping is its fast convergence to the desired precision. The number of neurons, hidden layers, and Fourier features can easily control the compression rate. The generated model also automatically returns an interpolated value for any point in the defined volume, thus allowing the user to reconstruct the data at any resolution.

Different model training approaches can be used. Multiple models can be trained separately for each variable, a single model can be trained for all variables at once, or a single model can be used for training with multiple time steps where structural similarities are expected.

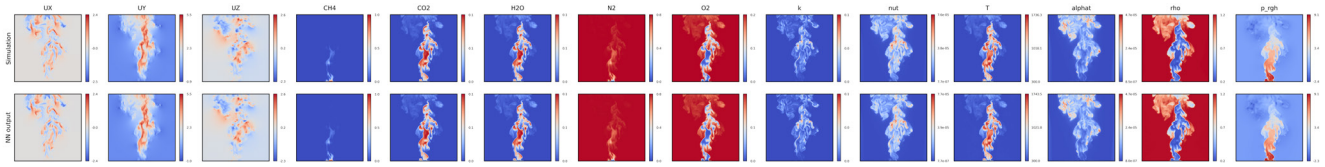


Fig. 1: Cross-section comparison of the simulation results (2.4GB) with the the neural network (26MB)

References

[1] M. Tancik, P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. T. Barron, R. Ng,. Fourier Features Let Networks Learn High Frequency Functions in Low Dimensional Domains. arXiv:2006.10739, (2020)

Poster:

Some Quirks of Using Shared Memory in CUDA Fortran

Ivan Vialov

VSC Research Center, TU Wien, Austria

Shared memory, a small amount of fast scratchpad space, which is used to cache accesses to global memory, is known to be crucial to achieve good performance in CUDA programming model. In the canonical example of matrix-matrix multiplication the peak performance can only be expected when the arithmetic intensity of the code is high enough, which implies a lot of accesses to shared, rather than global memory. Surprisingly, it is exactly this case where CUDA Fortran performs astonishingly poor losing 20% to 3.5x to CUDA C implementation of the same algorithms working with float/real data type.

In this work we provide test cases, which gradually increase the arithmetic intensity of GEMM and show how the performance stalls, instead of growing correspondingly. However, not all is lost and there is a simple, albeit somewhat obscure workaround: employing C-style indexing (starting with 0) of shared memory arrays, and building the code with `-gpu=nordc` restores the performance of CUDA Fortran to the level of CUDA C.

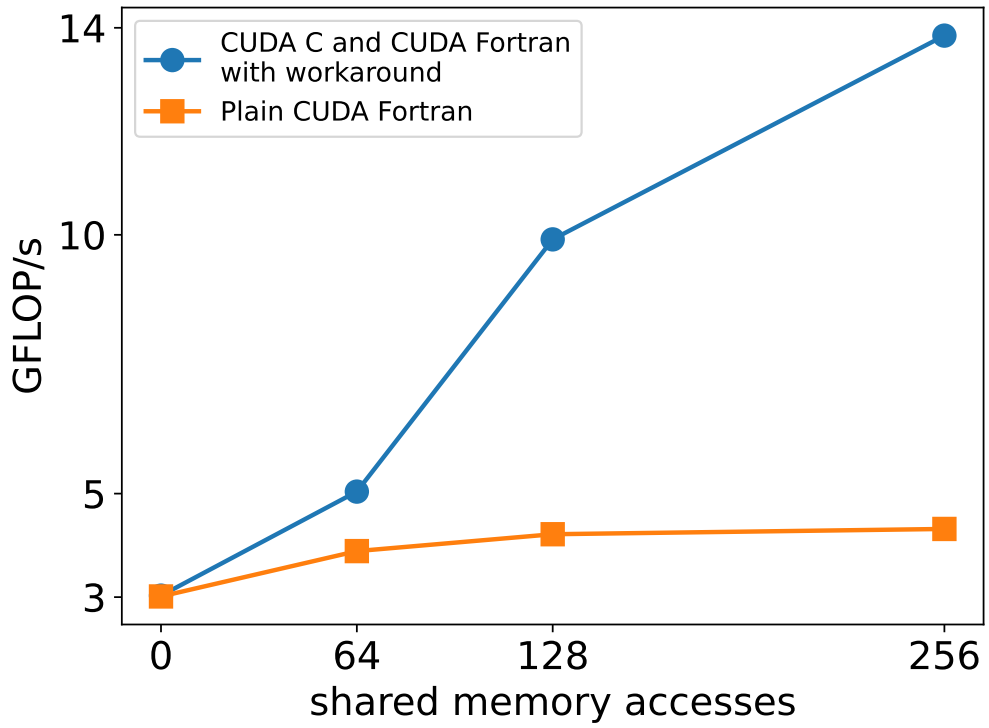


Fig. 1: Performance as a function of the number of shared memory accesses per thread needed to calculate one resultant matrix entry. The more the shared memory is reused the better the performance is expected to be, which is not the case for plain CUDA Fortran.

Poster:

Hyperparameter Optimization with Differential Evolution on Multiple GPUs

Teo Prica[a] **and Aleš Zamuda**[b]

[a] *IZUM - Institute of Information Science, Slovenia*
[b] *UM - University of Maribor, Slovenia*

In the poster, we will briefly present an experiment we carried out as part of our research studies. To perform the experiment, we used the largest Slovenian supercomputer EuroHPC Vega [1]. Thus, we prepared our deployment environment, including building a custom Singularity container for the PyTorch framework with required libraries. The composed code in programming language Python takes care of loading a dataset, setting up Distributed Data Parallel (DDP), functions for training, evaluation, and metrics, storing data in appropriate data frame, and graph plotting based on newly acquired data. DDP takes care of parallelism and exploitation of multiple nodes and GPUs and jobs are submitted through Slurm.

From a scientific understanding point of view, we are interested in how efficiently different algorithms perform, when we take into account their accuracy and consumption of computing resources. Data about the consumption of resources of individual jobs was obtained from Slurm database of completed jobs. The classification was performed using supervised Machine Learning (ML) through pre-trained models which are already available within the PyTorch framework. We used multi-label classification that allows each instance to be associated with multiple labels and differs from the conventional single-label classification, where each instance can be associated with a single label [2]. Since the main focus was not on data, we selected smaller publicly available dataset (CIFAR10). With such setup, we were able to apply fine-tuning to various aspects of hyperparameters that would take an effect [2], i. e., we examined the number of epochs and iterations, weights, learning rate (LR), batch size, and optimizers. As an optimizer, we could choose among popular approaches such as manual approach, Grid search, Random search, Bayesian optimization, automated tuning frameworks, specifically Differential Evolution (DE), or others. We chose the basic implementation of DE algorithm, as the algorithm itself is part of population algorithms and was introduced by Storn and Price in 1995 [3]. DE is successfully used in the field of optimization of numerical functions and applications to real problems within different domains [3]. The original DE has a main evolutionary loop through which, with the help of key and basic operators such as initialization, mutation, crossover, and selection, after initialization the final result is gradually improving by repeating the operators until termination [3]. In the evaluation phase of our experiment, we used some basic ML metrics such as accuracy, precision, recall, F_1 measure, epoch loss, weighted sum, metric aggregation, and a few others [2]. We wanted to answer, if it is worth running longer jobs for 2 or even less percent additional accuracy for double or more the job execution time of more complex and accurate models. As it turned out, while deploying individual jobs on multiple GPUs, through our experiment we were able to evolve and examine the efficiency of an ML algorithm according to accuracy and also report the consumption of computing resources.

References

[1] User documentation: hrefhttps://doc.vega.izum.si/https://doc.vega.izum.si/

[2] Han, M., Wu, H., Chen, Z., Li, M., Zhang X., Int. J. Mach. Learn. & Cyber. **14**, 697–724 (2023).

[3] Zamuda, A., 2020, Operacijske raziskave logističnih, transportnih in ekonomskih sistemov: zbrano gradivo. FERI – Faculty of Electrical Engineering and Computer Science.

Keynote Talk:

Building large language models: HPC users' perspective

Marko Robnik-Šikonja

Faculty of Computer and Information Science, University of Ljubljana, Slovenia

Large language models (LLMs) are at the forefront of recent successes in artificial intelligence (AI), with potentially large societal and technological impact. This caused widespread interest in their creation and adaptation to various scientific and commercial domains. Besides successful commercial LLMs, like ChatGPT, there are many open alternatives, like LLaMa-2, OLMo, and Aya, which enable their further pretraining and adaptation to specific domains and languages. However, building LLMs is an extremely data and compute hungry process, posing significant challenges to their creators and HPC infrastructures. As the web-crawled data is the main source of training data, the data quality problem is present for all languages. On the other hand, the data quantity problem is particularly grave for less-resourced languages, where sufficient amounts simply do not exist. Due to the need to optimize the LLM learning process and adapt it to the specific available hardware, the lack of HPC skills is also a widespread problem for AI researchers.

We present the background on modern LLMs architectures to explain their performance and assess their computational needs. Through a use-case of creating several LLMs for less-resourced languages, we identify challenges and opportunities in natural language processing and HPC computing. Further, we present the trends in LLMs, which show that they will soon require even more computing power, more texts, and also multimodal data such as images, speech, and video.

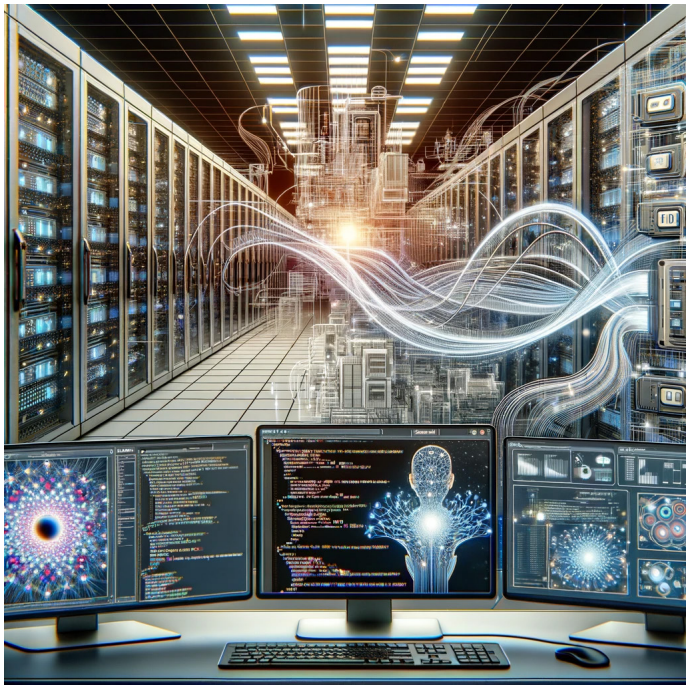


Fig. 1: LLMs need huge amounts of computing.

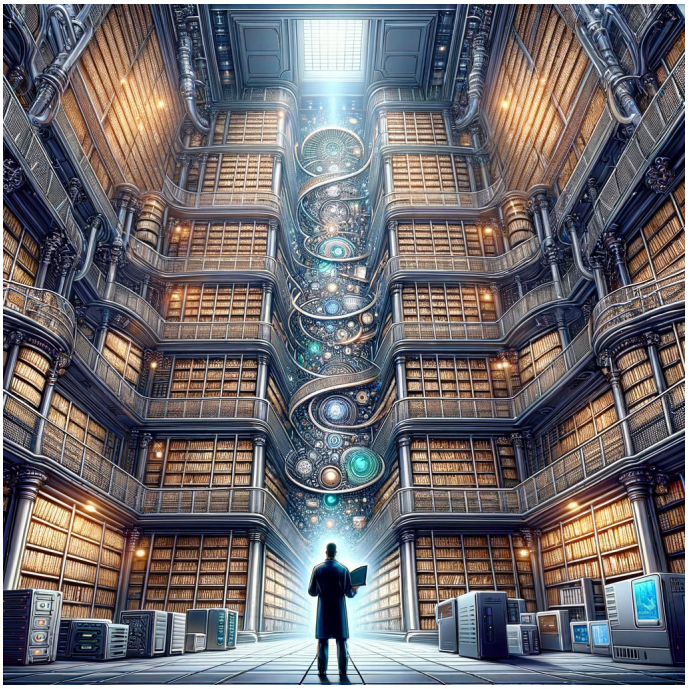


Fig. 2: LLMs require enormous amounts of textual data.

Randomised Optimisation Algorithms in DAPHNE

Aleš Zamuda

Faculty of Electrical Engineering and Computer Science, University of Maribor, Slovenia

Deploying Randomised Optimisation Algorithms (ROA) in DAPHNE [1] on EuroHPC Vega [2] allows benchmarking and research in novel and innovative models for Artificial Intelligence (AI) [3]. The methods that are supported in DAPHNE allow seamless distribution of AI memory that is required when an AI algorithm run requires a large memory that can be distributed across different HPC nodes. Using DAPHNE, the benchmarking can be not only run in the Runtime Environment [2] on an HPC that is much larger than a regular laptop computer, but also gather monitoring data of the workload while the algorithm is running, to obtain a benchmarking profile, allowing an informed scientific observation of a novel algorithm under test.

Figure 1 shows an example ROA run, with convergence plot for the HappyCat function, $f_1 = \left(\left(\sum\left(x^2,0\right)-10\right)^2\right)^{0.125} + \left(\sum\left(x^2,0\right)/2+\sum\left(x\right)\right)/10+0.5$ on a sample of different independent runs. Figure 2 shows a convergence plot for the function $f_2 = \sum\left(x^5+1+\max(x,0),0\right)$, also on 10 different independent runs. As seen from the plots, the fitness values are convergent, optimizing.

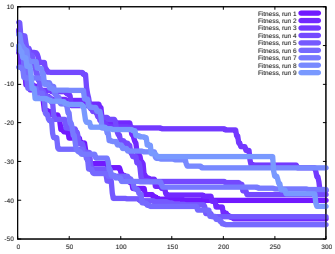


Fig. 1: Sample convergence on f_1 with DAPHNE ROA.

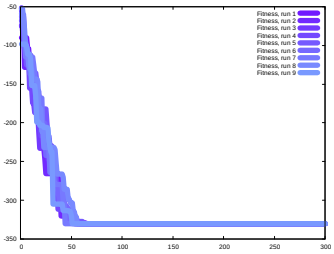


Fig. 2: Sample convergence on f_2 with DAPHNE ROA.

The results show an early demonstration of feasibility to compute ROA in DAPHNE. This initial implementation was run within DAPHNE v0.2. A linear time increase was observed with an increasing problem dimension size. The linear time increase is a good candidate for scaling and parallel execution of this workload, because the implementation uses matrix operations which are scalable in DAPHNE language over tiles of rows. Also, by changing the fitness evaluation function of the ROA, the system is applicable to various AI challenges like standard evolutionary computation test functions, or other more elaborated AI scenarios.

Further research can include applying optimization algorithms to larger problems, with larger and distributed memory and storage, benefitting from distributed processing with row-wise data in DAPHNE. Examples of these include vehicle navigation (e.g. Underwater Glider Path Planning — UGPP) and production (e.g. engineering design optimization), ROA algorithms self-design (e.g. through whole benchmark autoconfigurations), and more generative AI integration (such as Human Language Technologies — HLT).

Acknowledgement: this work is supported by EU project no. 957407 (DAPHNE) and ARIS P2-0041.

References

[1] Zamuda, A., High Performance, Edge And Cloud computing, Munich, Germany, January 17 – 19, 2024.

[2] A. Vontzalidis, S. Psomadakis, C. Bitsakos, M. Dokter, K. Innerebner, P. Damme, M. Boehm, F. Ciorba, A. Eleliemy, V. Karakostas, A. Zamuda, and D. Tsoumakos, 29th International European Conference on Parallel and Distributed Computing (Euro-Par), 28 August – 1 September 2023, Limassol, Cyprus.

[3] Zamuda, A., and Lloret, E., Journal of Computational Science **42**, 101101 (2020).

Evolutionary Implications of Multi-Scale Intelligence

Benedikt Hartl[a,b], **Sebastian Risi**[c], **and Michael Levin**[b,d]

[a] *Institute for Theoretical Physics, TU Wien, Austria*
[b] *Allen Discovery Center, Tufts, USA*
[c] *IT University of Copenhagen, Denmark*
[d] *Wyss Institute for Biologically Inspired Engineering, Harvard, USA*

The scientific community increasingly recognized the complex multi-scale competency architecture (MCA) of biology, comprising nested layers of competent agents forming the self-orchestrated substrate for the layer above, and, in turn, relying on the structural and functional plasticity of the layer(s) below [1]. The question of how natural selection could give rise to this MCA has been the focus of intense research. Here [2], we instead investigate the effects of such decision-making competencies of an MCA's agential components on the process of evolution based on *in-silico* neuroevolution experiments of developmental biology.

We specifically model the process of morphogenesis with neural cellular automata (NCAs) and utilize an evolutionary algorithm to optimize the corresponding neural network parameters of the uni-cellular agents with the objective of collectively self-assembling a two-dimensional spatial target pattern by regulating their cell types on the NCA's grid (see Fig. 1). Furthermore, we systematically vary the rate at which an NCA's uni-cellular agents can regulate their cell types during a noisy developmental stage, allowing us to continuously scale the agents' competency levels (indicated by the red knob in Fig. 1) from an MCA to a direct encoding scheme (zero competency).

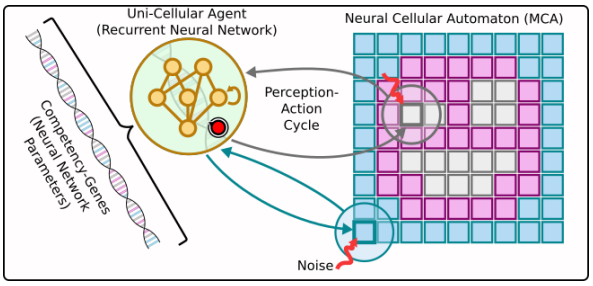


Fig. 1: Schematics of an MCA, whose genetically encoded uni-cellular agents self-assemble a target pattern via successive cell type updates.

With large-scale simulations, systematically scanning competency- and noise-levels in 75 statistically independent evolutionary searches at each parameter combination, we demonstrate that an evolutionary process can find solutions to such noisy pattern formation tasks significantly more efficiently (on average) when evolving the parameters of an MCA compared to evolving the target pattern directly (c.f., Fig. 2). Moreover, the evolved NCAs show strong signs of generalizability toward parameter changes and adaptability to modified objective functions of the evolutionary process.

Thus, the adaptive problem-solving competencies of the agential puzzle pieces in our NCA-based *in-silico* morphogenesis model strongly affect the underlying evolutionary process, suggesting profound evolutionary implications of the much more intricate MCA of biological life.

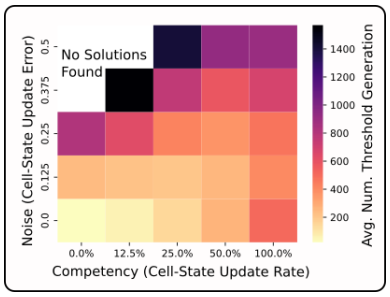


Fig. 2: The average number of generations of independent EA runs until a threshold generation is reached that solves the pattern-formation task per system parameter combination.

References

[1] Levin, M, CMLS **80**, 142 (2023).

[2] Hartl, B., Risi, S., and Levin, M., Proc. Royal Soc. B. (submitted Feb. 2024).

Zettaflops Computing: Why, when, and how?

Draško Tomić

Ruđer Bošković Institute, Croatia

There is a mounting evidence that today's even most powerful supercomputers are unable to solve a waste amount of problems in various fields of science and engineering. Some examples are protein folding in biochemistry, modeling the behavior of galaxies, understanding dark matter, and simulating the entire universe's evolution in cosmology, predicting the properties of new materials at the atomic and molecular scale with quantum mechanical calculations in material science, and simulating turbulent flows or the interaction of fluids with solid structures in fields like aeronautics and automotive design. There are many other examples in the fields of climate modeling, nuclear physics, and financial modeling where supercomputers, due to the lack of better performance, are unable to provide a solution in an acceptable amount of time. Besides and even though our computers are now much better than 15 years ago, they still malfunction frequently, thus preventing long running applications to finish successfully. In 2015 the paper [1] appeared showing that exploring bacterial biofilms is computationally extremely challenging and out of the reach at least until Zettaflop computers will be available. By extrapolating the performance development of the fastest supercomputers which increases exponentially over time, the author of this paper concluded that the first Exascale supercomputer will be available in 2018, and Zettaflop computers delivering 1000 times higher performance than Exascale computers will be available around 2028. Today we know that the first computer breaking Exaflop border was Fugaku which achieved 1.42 Exaflops with HPL-AI based benchmark in June 2020. So the arrival of the first Zettaflop computer will possibly be delayed and will happen no earlier than 2030. Quantum computing is emerging as a promising technology that will help us break the Zettaflop barrier. At first, quantum computers will be tightly linked to standard supercomputers and used as their co-processors for computationally intensive and very specific jobs, thus creating hybrid computer structures. Then, over time, they will evolve into fully independent units capable of solving general problems on their own. Quantum computing leverages the principles of quantum mechanics to perform computations using quantum bits, or qubits. Several mainstream technologies are being explored and deployed in the development of quantum computers. In this lecture, along with a comparative presentation of the basic technologies of quantum computing, experience will be shared about NAMD and Vini, applications that scientists from the Ruđer Bošković Institute, KBC Sestre Milosrdnice, and the University of Rijeka are running on HPC Vega and Lumi supercomputers within the frames of the EuroHPC project, and which in the future can benefit the most from Zettaflops, i.e., quantum computing.

References

[1] Tomić, D., "Exploring bacterial biofilms with NAMD, a highly scalable parallel code for complex molecular simulations," 38th International Convention on Information and Communication Technology, Electronics and Microelectronics, Opatija, Croatia, 2015, pp. 393-397, doi: 10.1109/MIPRO.2015.7160302.

Beyond Benchmarks: Comparing Parallel Programming Models in Real-World Scenarios

Philipp Gschwandtner[a] **and Ralf Kissmann**[b]

[a] *Research Center HPC, Department of Computer Science, University of Innsbruck, Austria*
[b] *Department of Astro- and Particle Physics, University of Innsbruck, Austria*

There are many parallel programming models in the HPC landscape. Some aim for high flexibility and responsibility of the programmer (e.g. MPI), while others advocate a separation of concerns, offering a domain-specific-language-like experience. These try to provide application developers with the means to express algorithms on a high level of abstraction and move performance optimization away to the system developer and toolchain (e.g. Celerity [1]). Regardless of their focus, all these models usually offer functional portability but are exposed to the problem of performance portability and programmer productivity. Many works study this topic, but most of them rely on evaluation using benchmarks, e.g. the NAS NPB, BLAS, Stream, other proxy apps or even textbook sample codes. Real-world applications however, especially with their engineering requirements, are often much more problematic. The few works that do consider real-world applications are usually limited to comparing one established implementation (MPI) with a new one meant to replace it (e.g. MPI+CUDA), for obvious effort reasons.

In this talk, we will discuss our aim to provide a qualitative study on performance portability and programmer productivity using a real-world application with interesting properties, Cronos, and porting it to multiple parallel programming models. A key idea is to employ computer science students that have some parallel programming background but are not experts, posing a case study resembling real-life circumstances and enabling the analysis of aspects such as ease-of-use, productivity, or documentation quality.

Cronos is an astrophysics 3D structured grid simulation developed and maintained at the University of Innsbruck with common but also challenging characteristics. While suitable for execution on GPUs, Cronos uses features that many programming models often struggle with including dynamic, data-dependent time step length calculation, posing a latency bottleneck, or parallel I/O using HDF5. Finally, we aim to keep a high standard in terms of software engineering, something that needs to be considered when integrating with parallel programming models. Cronos has been successfully used in a PRACE project on up to 16.000 cores on Joliot-Curie Rome at GENCI.

This project has been partially funded by EuroHPC-JU grant 956137 and the Vienna Scientific Cluster (VSC).

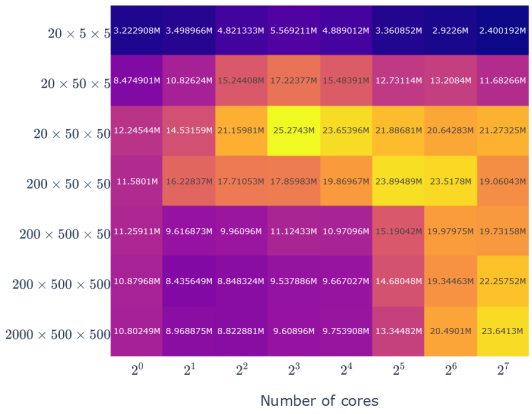


Fig. 1: OpenMP on VSC-5, cell updates/s for varying problem sizes and numbers of cores.

References

[1] Gschwandtner P., Kissmann R., Huber D., et al., "Porting Real-World Applications to GPU Clusters: A Celerity and Cronos Case Study," 2021 IEEE 17th International Conference on eScience (eScience), Innsbruck, Austria, 2021, pp. 90-98, doi: 10.1109/eScience51609.2021.00019.

Transparently Scaling Applications to Multiple GPUs with Celerity

Peter Thoman

Department of Computer Science, University of Innsbruck, Austria

HPC practitioners are commonly experts in a domain science, and seek to employ high-performance hardware in order to simulate or solve problems from their particular domain, e.g. a physical science. However, the current state of the art in HPC software assumes that everyone leveraging these systems is intimately familiar with their architectural details and programming intricacies. This has become even more challenging with GPU-based accelerators introducing additional complexities on top of the inherent difficulty of scaling any non-trivial application on distributed memory clusters. As a result, domain experts generally only have a few options, each with its own limitations: (i) they can work with existing software packages, which might be well-optimized but are limited to their existing set of algorithms and approaches; (ii) if available, they can try to use a high-level abstraction or skeleton framework for their domain; or (iii) they can spend a lot of effort on trying to program an HPC system with the available low-level tools such as "MPI+CUDA".

Celerity [1,2] is a declarative, dataflow-driven runtime system which aims to ease the burden of targeting distributed memory GPU clusters while enabling competitive performance. It provides a seemingly sequential API with data-parallel block processing similar to targeting a single GPU, which is expressed using the Khronos SYCL industry standard.

As illustrated in Fig. 1, based on the dataflow of a program formulated using this API, Celerity automatically builds a task and distributed command graph and ultimately schedules jobs onto accelerators.

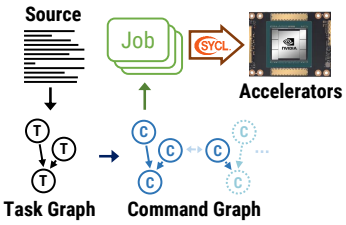


Fig. 1: Basic Execution Flow of a Celerity Program.

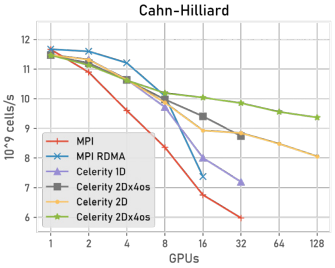


Fig. 2: Scaling experiments with a Celerity version of 3D stencil simulation compared to MPI baselines.

The Celerity runtime system takes over both work and data distribution across the GPUs in the system. Due to its in-depth knowledge of the application dataflow and capability of automatically maintaining data coherence, Celerity can be used to easily vary the distribution of data and transparently implement compute and communication overlapping. Experimentation across this space of data and work distribution enables achieving performance which is competitive with or even exceeds that of manual "MPI+X" implementations (see Fig. 2 for one scaling example).

In this talk, we will briefly introduce the Celerity API and its foundation, SYCL. We will explain the core working principles of its runtime system implementation, and explore the performance achieved across a variety of applications featuring distinct classes of parallelism.

References

[1] Salzmann, P., Knorr, F., Thoman, P. et. al. An Asynchronous Dataflow-Driven Execution Model For Distributed Accelerator Computing, IEEE 23rd Int Symp on Cluster, Cloud and Internet Comp (2023)

[2] Knorr, F., Thoman, P. et. al. Declarative Data Flow in a Graph-Based Distributed Memory Runtime System. Int J Parallel Prog 51, 150–171 (2023)

Optimising routines with sparse matrices on FPGA devices

Miha Fabčič and *Ratko Pilipović*

Faculty of Computer and Information Science, University of Ljubljana, Slovenia

Sparse matrices [1] are integral in diverse fields like scientific simulations, machine learning, and graph analysis, presenting unique challenges due to non-local memory accesses during operations. Compared to dense matrices, the non-local memory access can significantly impact the efficiency of algebraic computations on computer systems. To address this challenge, various compressed matrix notations, known as *compressed matrix formats*, have been proposed in the literature. The matrix compression formats aim to store matrices or enable efficient execution of sparse matrix operations by representing them compactly. Different notations prioritize minimizing memory footprint or enhancing computation efficiency, affecting the overall performance of sparse matrix operations.

Among sparse linear algebra routines, sparse matrix-vector multiplication (SpMV) stands out, facing challenges such as non-local memory access, leading to suboptimal cache utilization and limited computation parallelization. The optimization of SpMV is not only impacted by the chosen notation but also by the underlying architecture of computer systems, including the Central Processing Unit (CPU) and Graphics Processing Units (GPUs). In contrast to traditional systems, Field-Programmable Gate Arrays (FPGAs) offer a unique advantage by enabling the design of specialized computing units tailored for efficient SpMV operations.

Our research focuses on implementing accelerators on FPGA, which are custom-designed for various compressed matrix notations. We aim to identify the FPGA resources that significantly influence SpMV performance and determine the optimal accelerator architectures for different notations. Ultimately, we aim to determine the most effective compressed matrix notation or corresponding accelerator architecture for SpMV computations. Fig. 1 illustrates the delay of our initial Coordinate (COO)-based SpMV accelerator for different square matrix sizes.

Acknowledgments: This research was partially supported by Slovenian Research and Innovation Agency under Grants P2-0359 (National research program Pervasive computing) and under Grant BI-HR/23-24-009 (Bilateral Collaboration Project).

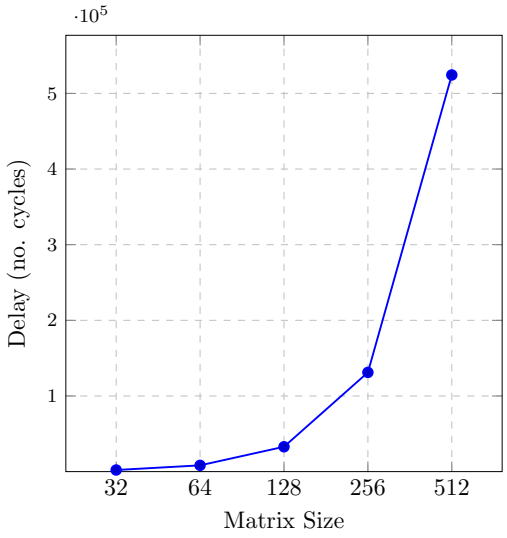


Fig. 1: Delay of COO-based SpMV accelerator for different matrix sizes

References

[1] Mohammed, T. and Mehmood, R., 2022. Performance Enhancement Strategies for Sparse Matrix-Vector Multiplication (SpMV) and Iterative Linear Solvers. arXiv preprint arXiv:2212.07490.

What You Always Wanted To Know About C++ Performance Portability (But Were Afraid to Do)

***Ruben Laso*, Diego Krupitza, and Sascha Hunold**

Faculty of Informatics, TU Wien, Austria

Performance-portable programming with ISO C++: With the advent of C++17, the Standard Template Library (STL) allows for the parallel execution of a large collection of algorithms, which makes the STL a viable framework for implementing performance-portable applications. In this talk, we introduce pSTL-Bench, a set of micro-benchmarks for analyzing the performance and scalability of several parallel STL implementations. This benchmark suite allows for comparing the performance of different compilers and backends on various parallel architectures, such as Intel or AMD CPUs and NVIDIA GPUs.

pSTL-Bench is a micro-benchmark suite to assess the performance of parallel algorithms from the STL. Currently, the suite includes micro-benchmarks for almost all parallel algorithms of the C++ standard library. Furthermore, it features a custom memory allocator to support the correct placement of the memory pages on NUMA systems [2].

Experimental Results: Fig. 1 presents the benchmark results for various compilers and backends. The experiments were carried out on one of the compute nodes of *VSC-5* and *Hydra* with 128 and 32 threads, respectively, matching the total core count of each compute node. Results are presented for different pairs of compiler frameworks and backends, e.g., GCC-GNU denotes the GCC compiler framework together with GNU's parallel STL implementation [1]. Generally, poor speedup was observed for several parallel algorithms, but in some scenarios, parallel implementations stand out, such as GNU's `sort` or the `for_each` implementations. We also experimented with GPU backends to determine the sweet spot for deciding whether to run a particular code on either a multi-core CPU or on a GPU. For small problem sizes, the overhead of using a GPU is too high to outperform the multi-core CPU code, but for medium problem sizes (see Fig. 2), parallel STL algorithms on the GPU outperform their counterparts on the CPU.

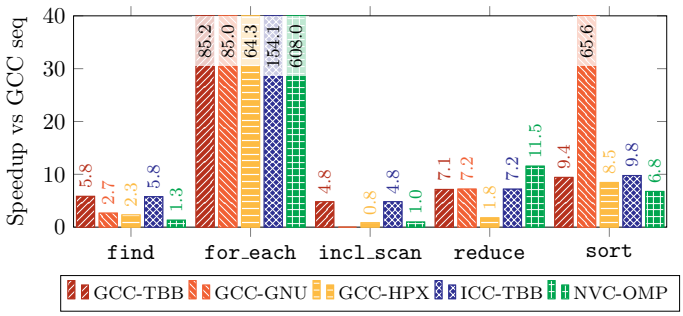


Fig. 1: Speedup on *VSC-5*. Problem size=2^{30} integers.

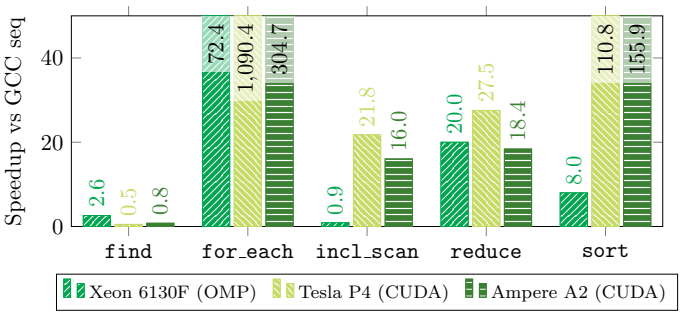


Fig. 2: Speedup on *Hydra*. Problem size=2^{25} integers.

References

[1] Singler, Johannes, Peter Sanders, and Felix Putze. "MCSTL: The Multi-core Standard Template Library." Euro-Par, 2007.

[2] Kaiser, Hartmut, et al. "HPX – The C++ Standard Library for Parallelism and Concurrency." Journal of Open Source Software 5.53 (2020): 2352.

Performance and Portability Assessment of CUDA and SYCL for Particle-in-Cell codes on different GPUs

Ivona Vasileska, **Pavel Tomšič, and Leon Bogdanović**

Faculty of Mechanical Engineering, University of Ljubljana, Slovenia

Heterogeneous computing, using CPUs and accelerators, has gained popularity for efficiently handling diverse workloads. This approach provides enhanced performance, energy efficiency and scalability, making it increasingly attractive for addressing modern computational demands. Nowadays, GPUs can be considered the dominant accelerator, and NVIDIA, Intel, and AMD are the most prominent manufacturers. This study assesses the performance and portability of CUDA and SYCL for the widely used plasma kinetic simulation code Particle-In-Cell (PIC). PIC codes use a dual-algorithm framework, with a particle pusher updating charged particle trajectories based on electromagnetic fields, and an electric field algorithm computing the field at each grid point through interpolation of particle contributions [1].

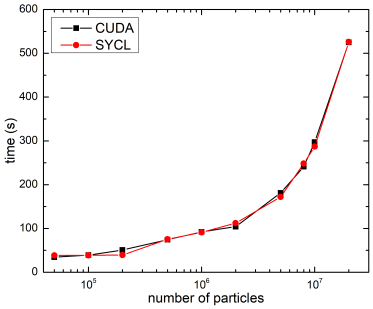


Fig. 1: Performance comparison between CUDA and SYCL

Using a 1D sheath simple PIC code, we have simulated various electron and ion numbers with temperatures of 3.0 eV and 1.0 eV, respectively, in a 0.01 m system with 10000 time steps on single GPUs [2]. Fig. 1 compares time simulation between CUDA (compiled with `nvcc` version 11.5.119) and SYCL (compiled with CLANG-LLVM compiler version 17.0.0) PIC codes on NVIDIA A100 GPU. Both show similar performance on NVIDIA A100, efficiently utilizing GPU architecture and leveraging optimized features for comparable execution speeds. Despite CUDA's limitation to NVIDIA hardware, we conducted a comparative analysis by running the SYCL PIC code (compiled with Intel oneAPI DPC++ compiler version 2023.2.0) on Intel GPU Max 1100 [3].

Fig. 2 results indicate that SYCL performs comparably on both NVIDIA and Intel GPUs with a small particle number, highlighting SYCL's efficiency with limited workloads. However, as particles increase from 1 to 20 million, a noticeable performance gap emerges, with Intel experiencing a twofold slowdown. This could be due to differences in GPU architectures, optimization strategies or memory bandwidth limitations impacting Intel's efficiency with larger workloads. As observed, SYCL scaling performance curve follows the matching curve of CUDA across various particle simulation scales, highlighting SYCL's portability advantages over CUDA, which is limited to NVIDIA GPUs. These findings showcase the potential opportunities SYCL brings to heterogeneous computing.

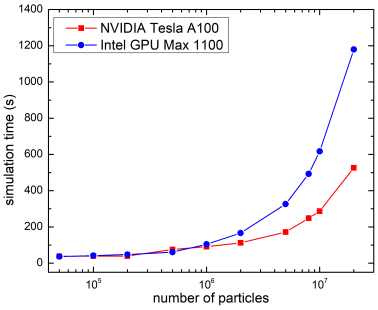


Fig. 2: Performance comparison between NVIDIA and Intel

References

[1] Tskhakaya, D. et al., "The Particle-In-Cell Method", Contrib. Plasma Phys. **47**, No. 8-9, 563–594 (2007).

[2] CUDA and SYCL PIC codes, https://bitbucket.org/lecad-peg/pic-gpus/src/main/.

[3] Intel DevCloud, https://www.intel.com/content/www/us/en/developer/tools/devcloud/services.html.

Keynote Talk:

Advancements, prospects, and challenges in weather and climate prediction: the rise of machine learning

Irene Schicker

Competence Unit Post-Processing, Department Analysis and Model Development, GeoSphere Austria, Austria

Since the 1950s, the digital revolution in weather forecasting has evolved starting with ENIAC and transitioning to the use of modern HPC resources in Numerical Weather Prediction (NWP) models and General Circulation Models (GCMs). This evolution has significantly improved both the forecast skill and the spatio-temporal resolution, crucial for decision-making across various sectors. NWPs and GCMs rely heavily on HPC infrastructure, are constrained by computational resources, physical parametrizations, and numerical stability limits. This has impacted further enhancements in resolution, data assimilation, and, consequently, forecast accuracy. While Machine Learning (ML) and Artificial Intelligence (AI) methods have been making strides in meteorology and climate science for applications such as pattern detection, targeted forecasts, and data analysis, they have only been applied in a view cases in a data-driven and hybrid way, not necessarily on a grid. In the last six years, however, data-driven uni/multivariate ML models started to emerge with models like Google's DeepMind GraphCast and ECMWF's AIFS [1], with significant potential in operational forecasting.

Recent developments include the first steps towards ML emulators of NWP models aiming to mimic operational models efficiently. Beyond weather forecasting, ML is increasingly applied in climate science for parametrization, downscaling [2], and post-processing [3]. The integration of ML with physical process models represents the next frontier in leveraging HPC for weather and climate forecasting, promising a broader application. This presentation will explore the latest ML advancements in weather and climate and the application of these innovations in practical weather and climate contexts.

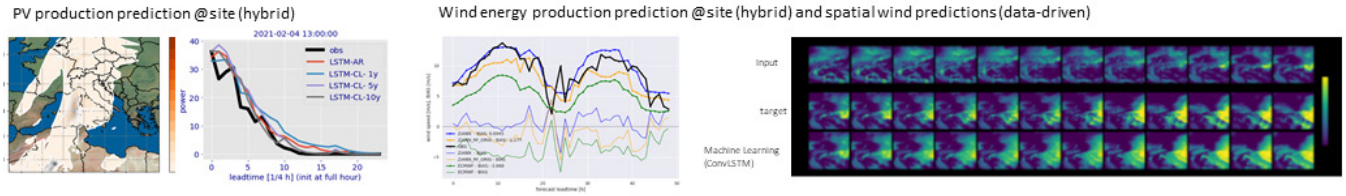


Fig. 1: Examples of data-driven and hybrid power production predictions using different types of strategies and ML models.

References

[1] Ben-Bouallegue, Z., *et al.*, The rise of data-driven weather forecasting (2023). https://arxiv.org/abs/2307.10128

[2] Amnuaylojaroen, T., Advancements in Downscaling Global Climate Model Temperature Data in Southeast Asia: A Machine Learning Approach, Forecasting, **6**(1), 1 (2024). https://doi.org/10.3390/forecast6010001

[3] Kiefer, S.M., Lerch, S., Ludwig, P., and Pinto, J.G., Can Machine Learning Models Be a Suitable Tool for Predicting Central European Cold Winter Weather on Subseasonal to Seasonal Time Scales?, Artificial Intelligence for the Earth Systems, **2**(4) (2023). https://doi.org/10.1175/AIES-D-23-0020.1

iCONtainer: climate and weather simulations with ICON in a singularity container

***Aiko Voigt*, Maximilian Meindl, and Michael Blaschek**

Department of Meteorology and Geophysics, University of Vienna, Austria

The Icosahedral Nonhydrostatic Weather and Climate Model (ICON) is a community model for the simulation of weather and climate across scales of space and time. ICON is used for operational weather forecasting by the German Weather Service, for climate projections of the Coupled Model Intercomparison Project, and for high-resolution simulations that form the physical basis of digital twins of the Earth. ICON is also used for education and training of students, for example as part of the MSc Meteorology course "Climate Modeling Lab" at the University of Vienna. However, porting ICON to high-performance computing systems and using it efficiently on these systems can be quite challenging. This challenge is exacerbated by the fact that climate model simulations require a software stack that remains stable over several years, which is often not the case at multi-purpose HPC systems. In addition, many small university research teams lack the resources and expertise that are required for model porting.

With iCONtainer, we address these challenges by running ICON in a singularity container. Because of the long legacy of using ICON with Intel, iCONtainer uses the Intel compiler suite and Intel-MPI.

Our presentation will cover the design choices of the container and our experience with it on several European HPC systems: the Vienna Scientific Cluster VSC4 and VSC5, the LUMI system of the EuroHPC Joint Undertaking, and the Levante system of the German Climate Computing Center. We will consider two drastically different flavors of ICON: a coarse-resolution version with 100 km grid spacing (ICON-ESM; [1]), and a high-resolution version with km-scale grid spacing (ICON-Sapphire; [2]). Because the two flavors differ drastically in their grid spacing, they sample different complexities and challenges. We will compare the model performance for the container version and the native installation on these HPC systems, showing that the performance penalty of the container ranges from small to acceptable. We will further show how the container version avoids some of the model problems that we encounter with native installations. Finally, we will describe remaining issues, such as those related to finding the best match between the MPI libraries of the host HPC systems and the container.

While its development is ongoing, we plan to make iCONtainer available to the public. By making it easier for domain scientists to use the state-of-the-art climate and weather model ICON, we hope that iCONtainer will help to democratize climate and weather modeling.

References

[1] Jungclaus, J. H., Lorenz, S. J., Schmidt, H., Brovkin, V., Brüggemann, N., Chegini, F., et al., Journal of Advances in Modeling Earth Systems, **14**, e2021MS002813 (2022).

[2] Hohenegger, C., Korn, P., Linardakis, L., Redler, R., Schnur, R., Adamidis, et al., Geoscientific Model Development, **16**, 779 (2023).

Enabling Global Flood Monitoring with SAR Datacubes and the Vienna Scientific Cluster

Mark Edwin Tupas[a,b], **Florian Roth**[a], **Bernhard Bauer-Marschallinger**[a], **Felix Reuß**[a], **Bernhard Raml**[a], **and Wolfgang Wagner**[a,c]

[a] *Remote Sensing Research Group, Department of Geodesy and Geoinformation, TU Wien, Austria*
[b] *Department of Geodetic Engineering, University of the Philippines Diliman, Philippines*
[c] *EODC Earth Observation Data Centre for Water Resources Monitoring, Austria*

Floods affect a significant portion of the global population annually, ranking high among the costliest and deadliest natural disasters. Mapping floods with earth observation satellites offers substantial advantages over field-based methods, contributing to their pervasive use. Since 2021, the Remote Sensing Group of TU Wien GEO has been part of the Copernicus Global Flood Monitoring (GFM) project. We contribute our Synthetic Aperture Radar (SAR) datacube-based flood mapping algorithm [1] to an ensemble workflow using Sentinel-1 satellite images to generate flood maps globally in near-real-time.

Among other issues, GFM operations must overcome timeliness and robustness concerns. The former requires algorithms to run expediently, while the latter entails good performance across varying terrain and climate types. Our research addressing these concerns is enabled by the Earth Observation Data Centre for Water Resources Monitoring (EODC) infrastructure and the Vienna Scientific Cluster (VSC).

The timely execution of the near-real-time processing chain relies on several global SAR time-series statistics and parameters pre-processed ahead of time. These intermediate datasets are generated annually in bulk. We give an overview of the different parameter processing chains and their challenges, also presenting the details of the input Sentinel-1 datacube and the Python-based software used in the context of High-Performance Computing. To illustrate the performance, one workflow is presented in detail, the updated global harmonic parameter computation processing chain [2].

Furthermore, we will show results from extensive parameterization and optimization experiments on the VSC that were carried out to ensure that our algorithm is robust. Most SAR-based flood mapping research demonstrates algorithms on select flooding events, often neglecting pitfalls relevant in automated operations, such as false alarms in non-flooded conditions. To balance flood mapping performance with robustness to such false alarms, our research encompasses multiple study sites and significant time-series chunks [3]. We will discuss exemplar experiments leveraging the VSC.

References

[1] Bauer-Marschallinger, B., Cao, S., Tupas, M. E., Roth, F., Navacchi, C., Melzer, T., Freeman, V., & Wagner, W. , Satellite-Based Flood Mapping through Bayesian Inference from a Sentinel-1 SAR Datacube. Remote Sensing, 14(15), Article 15 (2022).

[2] Tupas, M., Navacchi, C., Roth, F., Bauer-Marschallinger, B., Reuß, F., & Wagner, W. , Computing Global Harmonic Parameters for Flood Mapping using TU Wien's SAR Datacube Software Stack. The Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci., XLVIII-4-W1-2022, 495–502 (2022).

[3] Tupas, M. E., Roth, F., Bauer-Marschallinger, B., & Wagner, W. , Improving Sentinel-1 Flood Maps Using a Topographic Index as Prior in Bayesian Inference. Water, 15(23), Article 23 (2023).

MUSICA (MUlti-SIte Computer Austria)

Elias Wimmer[a], **Martin Thaler**[b], **Markus Hickel**[a], **and Adam McCartney**[a]

[a] *VSC Research Center, TU Wien, Austria*
[b] *ZID, University of Innsbruck, Austria*

In the project MUSICA (MUlti-SIte Computer Austria), funded by FFG [1] within the framework of Quantum Austria [2] and additionally funded by the Austrian Ministry of Education, Science, and Research (BMBWF) [3], we got the chance to build a truly distributed HPC platform for science and research. In our presentation we will discuss the advantages and challenges of such a complex system. Especially, creating a common user base and achieving consistency of data. One of our primary goals with MUSICA is to design a system that offers the user "cloud-style" composability, meaning users can build their own services using a predefined set of components, rather than trying to provide a certain set of services, that attempts to satisfy our increasingly diverse user requirements. Especially in the rapidly developing field of artificial intelligence (AI), we saw the need for new ways to access HPC resources in addition to traditional batch computing.

To accommodate demanding AI workloads, the systems will provide a large number of the newest Nvidia H100 GPUs. The systems will be located across three universities in Austria, University of Innsbruck, Johannes Kepler University Linz, and at TU Wien. In order to make data easily accessible at all sites, the system will use a dedicated high-speed network to transfer the data to the site specified by the user. Our goal is to offer users a seamless experience of using the system, irrespective of location. In addition to identical hardware, this will be achieved by providing a common software stack and unified identity management at all sites.

System Specification

Innsbruck & Linz

- 80 x GPU nodes
  4 x Nvidia H100 94 GB, 2 x AMD Epyc 9654, 96 cores and 768 GB memory, 4 x NDR 200
- 48 x CPU nodes
  2 x AMD Epyc 9654, 96 cores and 768 GB memory, 4 x NDR 200
- 4 PB all-flash scratch storage
  WekaFS, up to 1800 GB/s read, 750 GB/s write performance

Wien

- 112 x GPU nodes
  4 x Nvidia H100 94 GB, 2 x AMD Epyc 9654, 96 cores and 768 GB memory, 4 x NDR 200
- 72 x CPU nodes
  2 x AMD Epyc 9654, 96 cores and 768 GB memory, 4 x NDR 200
- 4 PB all-flash scratch storage
  WekaFS, up to 1800 GB/s read, 750 GB/s write performance

References

[1] https://projekte.ffg.at/projekt/4496688

[2] Quantum Austria is financed by the EU Recovery and Resilience Facility https://commission.europa.eu/business-economy-euro/economic-recovery/recovery-and-resilience-facility_en

[3] https://www.bmbwf.gv.at/

Cython vs. Numba vs. Mojo: A Comparison of Different Approaches to speedup Python Language Execution

Benjamin Hackl

Department of Mathematics and Scientific Computing, University of Graz, Austria

For several years now, Python has been defending its position as one of the most popular programming languages (according to, for example, the TIOBE index, or the Stack Overflow developer survey). While there are numerous advantages of the interpreted nature of Python, a major drawback of its design is the performance aspect.

In HPC applications, plain Python (and to be more specific, its CPython incarnation) is often only used for rapid prototyping (leveraging its simplicity) – but as soon as the priority shifts to making computations more performant, implementations in classical compiled languages like C++ are preferred. Naturally, this prompts the question whether it is somehow possible to take the existing (plain) Python prototype and *make it faster*. Three substantially different approaches to this problem are realized by the following projects:

Cython. As per the description in the project's documentation, "Cython is Python with C data types". The Cython compiler takes a piece of (compatible) Python code and converts it to C code, which is then compiled into an extension module that Python code can interact with. Cython supports native parallelism and integrates well with `NumPy`, a fundamental package for (high-dimensional) numerical computations.

Numba. This project implements a "JIT compiler that translates a subset of Python and `NumPy` code into fast machine code". Practically, this mechanism is particularly convenient for speeding up existing code as only the `@jit(nopython=True)` decorator needs to be added—the resulting performance does depend strongly on the eligibility of the code. Numba is built on top of the LLVM compiler infrastructure.

Mojo. Having made its first appearance only in 2023, this project is by far the youngest of the three. It aims at implementing a new superset-language of Python (stated as a long-term goal) that is primarily designed to leverage the relatively new compiler development framework MLIR [2], which is particularly well-suited for new and/or non-traditionally designed CPUs and GPUs. In contrast to Cython, Mojo aims to only invoke CPython to ensure compatibility with existing Python code. Whenever this compatibility layer is not required, it solely relies on its own MLIR-based infrastructure to compile and run code.

We take a classical mathematical problem from image processing, the minimization of the total variation of an image in the context of denoising [1], which we solve using a simple Python implementation. Then, we compare the effort required to apply each of the speedup approaches discussed above as well as the resulting performance.

References

[1] Chambolle, Antonin, An algorithm for total variation minimization and applications, J. Math. Imaging Vision **20**, 89 (2004).

[2] Lattner, Chris et al., MLIR: Scaling Compiler Infrastructure for Domain Specific Computation, 2021 IEEE/ACM International Symposium on Code Generation and Optimization (CGO).

How much memory per CPU core is requested?

***Alois Schlögl*, Waleed Khalid, Stefano Elefante, and Stephan Stadlbauer**

Institute of Science and Technology Austria (ISTA), Klosterneuburg, Austria

Investments in HPC infrastructure are expensive, and it is important to specify the technical requirements according to the actual need. The amount of available memory per CPU core [1] is one of these crucial parameters. Usage data from the past has been used to compare requested with currently available resources.

The usage data was extracted from the schedulers database (Slurm) for a 10-month time span. During that period, the ISTA scientific compute cluster has been used by 242 users from 47 research groups and about 7.6 Mio Jobs were processed, using 19 million (virtual) core hours .

The requested memory per core was evaluated for each job, and used to compute the cumulative distribution of the workload (i.e. core-hours) of each job. Moreover, the available memory per (virtual) core of each machine was extracted, and the cumulative distribution of the memory (total 76 TB) is obtained. The cumulative distribution for these two metrics are shown in Fig. 1, and the corresponding quantile values are extracted in Table 1.

Table 1: Quantile values of available and used memory [MB] per core.

quantile	available	used
0.10	2584	250
0.25	3815	1000
0.50	3815	3000
0.75	7629	6250
0.90	15625	11000

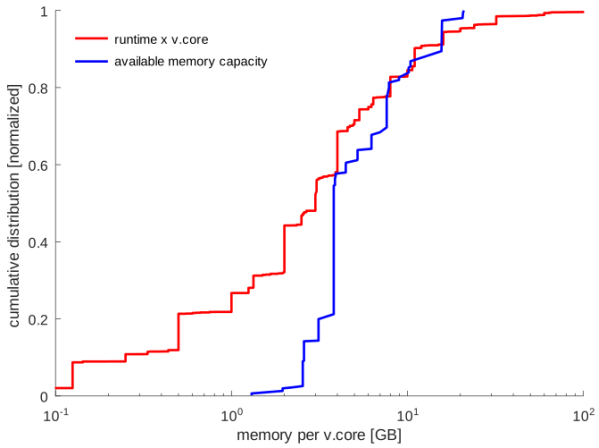


Fig. 1: Cumulative distribution of memory per core.

This HPC cluster is rather heterogeneous with nodes ranging from 1.3 to 20 GB per core. The median of requested memory-per-core is 3 GB per core, with an inter-quartile range between 1 and 6.25 GB. About 5% of the workload would benefit from more memory, and about 25 % of the workload requires 1 GB per core or even less. Overall, the available hardware does match reasonably well to the workload distribution, and will be used for deciding on the technical specification of future extensions.

The actual use and the installed capacity in terms of memory per core have been compared. These results were useful for specifying the requirements of the next extension of the HPC cluster at ISTA.

References

[1] Darko Zivanovic, et al. (2017) Main Memory in HPC: Do We Need More or Could We Live with Less? ACM Transactions on Architecture and Code Optimization, Vol 14(3) pp.1–26.

Automating Access to OpenID Connect Services from Linux Systems

Wiktor Nastał

Wrocław University of Science and Technology, Poland

This presentation introduces an innovative solution designed to automate the process of obtaining credentials from services authenticated through OpenID Connect in the headless Linux environment. The main goal of this solution is to empower both users and administrators to execute diverse workflows on high-performance computing resources. Securely storing and effectively managing access tokens within the user profile is achieved via oidc-agent [1]. This enables controlled access to various services, ensuring the tokens are kept secure. Moreover, the obtained tokens can be utilized for automating configurations across multiple services. This solution places a strong emphasis on obtaining user consent, ensuring that the automation process aligns with privacy and security standards. Users have the ability to grant explicit consent for the utilization of access tokens, thereby maintaining transparency and control over their credentials. This user-centric approach not only enhances security but also builds trust in the automation process. The MyToken [2] service provides a centralized and user-friendly platform for overseeing and customizing access privileges associated with users' tokens.

The process begins with the deployment of oidc-agent on any system, establishing seamless communication with MyToken. Acting as a intermediary, MyToken retrieves access tokens from the Identity Provider and then efficiently transmits them back to oidc-agent. This facilitates the straightforward automation of accessing OpenID Connect services, such as cloud storage solutions, secure vaults, and other essential services through the system where oidc-agent is deployed.

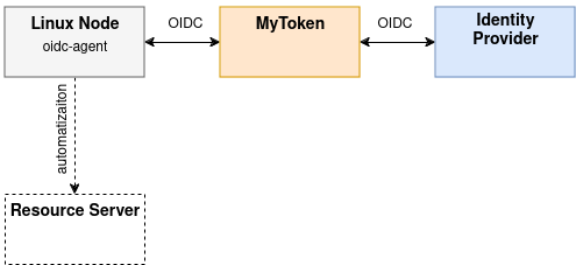


Fig. 1: Access flow with oidc-agent and MyToken.

Users have full control over their tokens through the MyToken service. This capability allows users to exercise precise control over the access tokens obtained from the Identity Provider, contributing to the establishment of a robust and secure token management system. Additionally, monitoring and oversight of data can be conducted within the services themselves due to the clear identification of the user during the automation process.

Throughout the presentation, we will explain the operational framework of this solution, highlighting crucial stages of the process. Additionally, a practical utilization examples will be showcased to illustrate the advantages of implementing this solution in the context of secure and privacy-compliant access administration within the HPC domain.

References

[1] Zachmann, G., OIDC-Agent: Managing OpenID Connect Tokens on the Command Line (SKILL 2018-Studierendenkonferenz Informatik) (2018).

[2] Zachmann, G., Mytoken-OpenID Connect Tokens for Long-term Authorization (Master's Thesis, Karlsruher Institut für Technologie (KIT)) (2021).

Accelerating Fortran Codes: Merging Intel Coarray Fortran with CUDA and OpenMP

James E. McKevitt[a], **Eduard I. Vorobyov**[a], **and Igor Kulikov**[b]

[a] *Department of Astrophysics, University of Vienna, Austria*
[b] *Institute of Comp. Math. and Mathematical Geophysics SB RAS, Russia*

Fortran's prominence in scientific computing requires strategies to ensure both that legacy codes are efficient on high-performance computing systems, and that the language remains attractive for the development of new high-performance codes. We demonstrate a novel and robust integration of Intel Coarray Fortran (CAF) — for distributed memory parallelism with familiar syntax — with Nvidia CUDA Fortran for GPU acceleration and OpenMP (OMP) for shared memory parallelism. We applied this to a nested grid potential solver [1], used in a protoplanetary disk code [2], showcasing significant performance improvements, comparable to those achieved with the Message Passing Interface (MPI) but while retaining Fortran syntax [3].

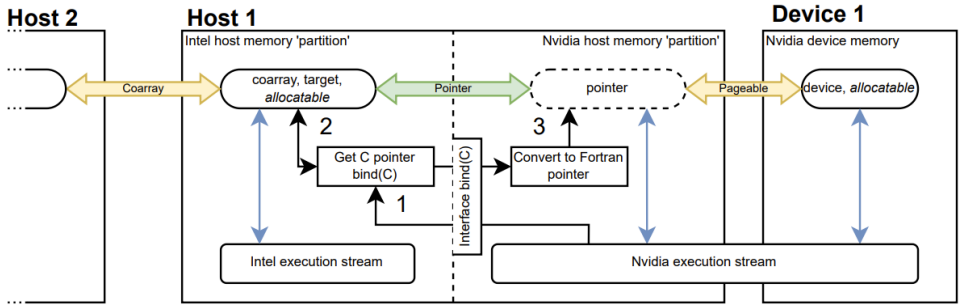


Fig. 1: The setup procedure required to use Intel CAF with Nvidia CUDA Fortran.

We consider three main aspects: how to best manage pageable and pinned memory to speed up transfers between CPU and GPU memory; how to optimise CPU-GPU affinity, considering the VSC's NUMA architecture; and how C-pointers and C-bound subroutines can robustly interface the two compilers. We also discuss the limitations of our approach, and compare its performance with MPI through weak and strong scaling tests.

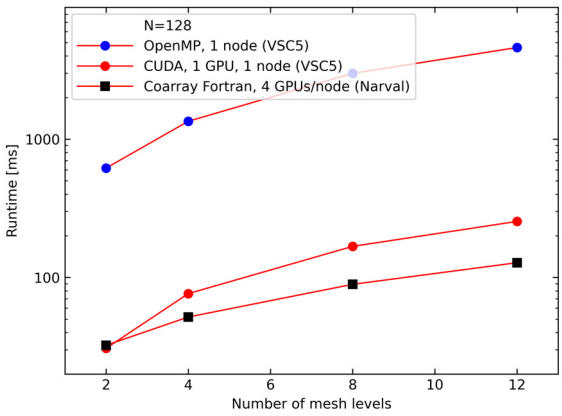


Fig. 2: Scaling with increasing parallelism.

Acknowledgements: This work was supported by the FWF project I4311-N27 (J. M., E. I. V.) and RFBR project 19-51-14002 (I. K.). Simulations were performed on the Vienna Scientific Cluster (VSC) and on the Narval Cluster provided by Calcul Québec and the Digital Research Alliance of Canada.

References

[1] Vorobyov E.I., McKevitt J.E., Kulikov I. and Elbakran V., A&A **671**, A18 (2023).

[2] Vorobyov E.I., Kulikov I., Elbakran V. and McKevitt J.E., A&A, Accepted.

[3] McKevitt, J.E., Vorobyov E.I., and Kulikov I., J. of Parallel and Distributed Computing (Submitted).

Keynote Talk:

Cell Detection for the 2020s – Computational Pathology and HPC Perspective

Dejan Štepec

Faculty of Computer and Information Science, University of Ljubljana, Slovenia

Background: The digitization of histopathological diagnostics has advanced considerably in the last decade, mainly due to the advent and increased availability of scanning devices for diagnostic tissue samples, which allows diagnostics that is comparable to traditional microscope-based diagnostics. The digitization of tissue samples and the simultaneous development of the field of artificial intelligence (AI) and computer vision have provided the possibility to develop AI-assisted tools to assist expert pathologists with diagnostics, as well as in drug discovery efforts. Many such tools have been successfully applied lately in various fields of pathology, where their diagnostic comparability with pathologists' has been demonstrated [1, 2].

Problem Statement: We address the fundamental problem of cell (or other biological structures) detection and classification which represents the basis for various diagnostic assessments, as well as drug discovery efforts. Developing automated methods to tackle this problem requires extensive annotation effort: labeling cancerous regions and thousands of cells of interest. Furthermore, the processing of digitized gigapixel imagery requires robust computer vision methods capable of scoring in difficult conditions: (1) with the lowest possible amount of labeled data, in terms of quantity and complexity and (2) under large appearance variability, that is naturally present in samples due to the chemical processes, morphological variability of the tissue, and characteristics of scanning devices.

Contribution: We introduce a **Patherea** framework that enables robust cell detection and classification with point-based annotations only. The framework also enables the utilization of easily available unlabeled digitized samples for self-supervised pre-training. The processing of digitized gigapixel tissue samples and the building of pathology foundational models requires extensive usage of HPC resources. We introduce the largest to date manually labeled dataset and demonstrate the improvements of our model on existing public datasets.

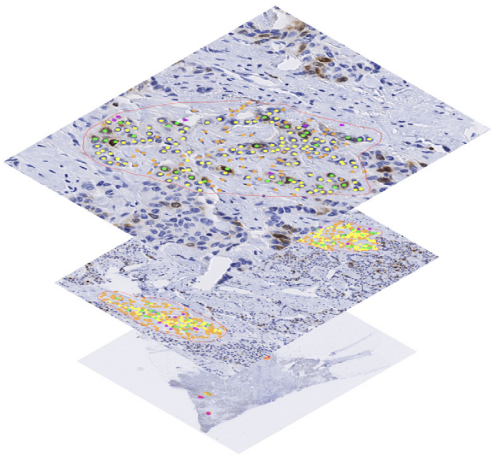


Fig. 1: Patherea dataset example.

References

[1] Gupta, A., *et al.*, SegPC-2021: A challenge & dataset on segmentation of Multiple Myeloma plasma cells from microscopic images, Medical Image Analysis, **83**, 102677 (2023).

[2] Štepec, D., and Skočaj, D., Unsupervised Detection of Cancerous Regions in Histology Imagery using Image-to-Image Translation, Proceedings of 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 3780 (2021).

Implicit Solvation Machine Learning Model for Molecular Simulations of Ionic Media

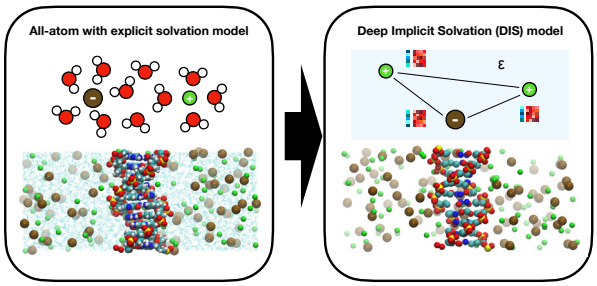
Amaury Coste[a], **Ema Slejko**[a,b], **Julija Zavadlav**[c], **and Matej Praprotnik**[a,b]

[a] *Laboratory for Molecular Modeling, National Institute of Chemistry, Slovenia*
[b] *Department of Physics, Faculty of Mathematics and Physics, University of Ljubljana, Slovenia*
[c] *Professorship of Multiscale Modeling of Fluid Materials, TUM School of Engineering and Design, Technical University of Munich, Germany*

In the past decade, progress in massively parallel processor hardware has led to an incredible increase in computational resources. It enables the development of novel applications in machine learning and molecular modelling research fields.

All-atom molecular dynamics (MD) is a powerful and widely used computational technique for understanding and predicting the physical behaviour of systems at an atomistic level. However, since the computational cost scales with the number of atoms, the method is often inapplicable to biologically relevant time scales and system sizes. By treating a part of the system, typically solvent, implicitly, the models are computationally efficient since the solvent could represent 90% of the simulation box. Nevertheless, the MD simulations of biophysical systems require accurate modelling of their native environment, *i.e.*, aqueous ionic solution, as it critically impacts the structure and function of biomolecules. Another crucial criterion regarding the simulation of ions and highly charged molecules are the electrostatic interactions.

Here, we present the deep implicit solvation (DIS) model for sodium chloride solutions that satisfies the requirements [1]. The delta learning approach is employed where the physics-based potential captures the long-range interactions. Owing to the machine learning (ML) potential, the model can capture the many-body potential of mean force, while the implicit water treatment renders the model inexpensive. The chosen ML potential is based on a strictly local equivariant neural network architecture.



Our DIS model showcased excellent accuracy for the structural properties of aqueous salt solutions with concentrations ranging from 0.15 to 2.0 mol L^{-1}. The molecular system including a DNA molecule likewise confirmed that the model can simultaneously capture the effective ion interaction in two distinct environments: close to the DNA molecule and in bulk. By using the computing capacities of current GPUs to train machine learning models, we can obtain a model for nonbonded interactions that can capture the many-body potential of mean force. Thus, the resulting model is computationally efficient and enables scaling up the simulations. This work paves the way for efficient simulations of biomolecules in an ionic media by using an implicit solvation model. It can also be applied to study other systems such as solid–liquid interfaces.

References

[1] Coste, A., Slejko, E., Zavadlav, J. and Praprotnik M., J. Chem. Theory Comput. **1**, 411 (2024)

OpenFOAM CFD simulation performance and scalability analysis

Kirils Surovovs **and Jānis Virbulis**

Institute of Numerical Modelling, University of Latvia, Latvia

The `OpenFOAM` library is widely used for high-performance computing (HPC) in both academical and industrial research groups. Due to large mesh sizes – several million or more elements – parallel computing often is inevitable for computational fluid dynamics (CFD). However, the higher the number of the used cores, the more resources are spent on communication between processors, data transfer and other operations. It is important to ensure that the limited resources are used efficiently.

Previously the CFD scalability tests were performed on relatively small clusters that are located in Latvia (using *Intel(R) Xeon(R) Gold 5220R* and *6140* processors) [1]. However, the time spent on communication increased so much when multiple nodes were used, that the number of finite volume operations per second did not increase at all. The present work shows the results of the tests performed on the Luxembourg supercomputer `MeluXina` (12th most powerful supercomputer in the EU, *AMD EPYC 7H12 64-Core* processors) [2]. An example of the obtained results is the strong scalability curve shown in Fig. 1.

There are several ways to describe the efficiency of HPC simulations. One of them is using the metrics proposed by the Performance Optimisation and Productivity center [3]. These metrics include not only global efficiency (which can be evaluated simply by running a simulation and measuring its execution time), but also allow to analyze its components. Some examples are parallel efficiency (how well the computational load is balanced between threads) and computation scalability (how much the number of instructions increases).

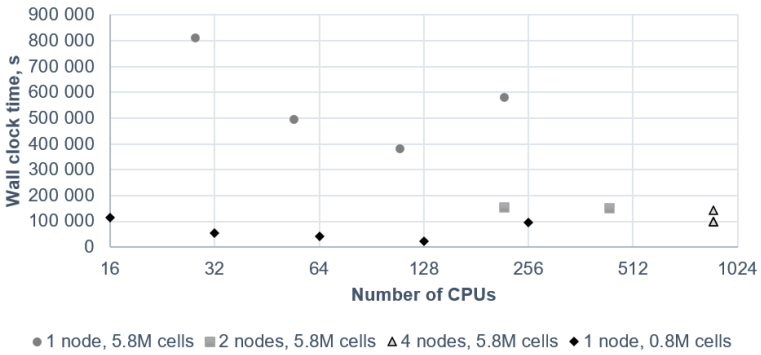


Fig. 1: Strong scalability curve obtained on `MeluXina` cluster with fine (5.8 million cells) and coarse (0.8 million cells) meshes.

The simulations on `MeluXina` cluster showed better scalability than on the previously tested clusters. An especially good result is, for example, almost 14-fold speed-up with the mesh with 3.2 million cells, when core number increased from 2 to 16. Such speed-up is superlinear (the speed increases more than the number of cores) and it demonstrates the efficient use of HPC resources. However, in most of the tested cases the efficiency dropped significantly when more than two nodes were used.

References

[1] Surovovs, K. and Virbulis, J., 81st International Scientific Conference of University of Latvia: Digital Twins for Sustainable and Safe Living Environment, 8 (2023).

[2] https://luxembourg.public.lu/en/invest/innovation/meluxina-supercomputer.html

[3] Boyle, J., POP Performance Analysis Methodology Workshop, 1 (2021).

Accounting and efficient use of GPUs in a slurm based HPC cluster

Johann Dorn[a,b], **Jürgen Hoffmann**[b,c], **Michael Neumayer**[a,b], and ***Thomas Rattei***[a,b]

[a] *Centre for Microbiology and Environmental Systems Science, University of Vienna, Austria*
[b] *Life Science Compute Cluster, University of Vienna and Medical University of Vienna, Austria*
[c] *Department of Structural and Computational Biology, University of Vienna, Austria*

Graphical processing units (GPU) are used by an increasing number of scientific programs in Life Sciences. A typical example is AlphaFold, a computational method that predicts protein structures with atomic accuracy even in cases in which no similar structure is known [1]. Most High-Performance Computing (HPC) infrastructures therefore need to increase the number of GPU nodes. The high costs for GPUs make detailed monitoring and efficient scheduling of GPUs necessary for economic operation and growth of HPC systems.

The Life Science Compute Cluster (LiSC) uses the job scheduler slurm. Slurm allows scheduling of GPUs as generic resources in two ways: exclusive use and shared use. However, in contrast to CPU and memory, slurm currently doesn't provide accounting data for the actual use of GPU cores and GPU memory of jobs. Based on nvidia-smi and control groups, We have therefore developed a custom software for the monitoring and statistical accounting of GPU use in slurm.

The GPU accounting resulted in valuable usage data. The data are summarized in the LiSC dashboards (Fig. 1). Per job data are stored as comment in the slurm accounting database and are sent as text summary with the job end mails. Most GPU-enabled software at LiSC use GPUs only for short periods during their runtime. A typical example is again Alphafold, which typically uses GPUs during about 1 percent of the job runtime only. The results overall indicated that most GPU-enabled software in Life Sciences requires a high CPU:GPU ratio and allows shared GPU use by default. Slurm at LiSC has been configured and users have been instructed accordingly. Upcoming hardware extensions of LiSC have therefore been planned as mixed nodes with 2 CPUs and 1 GPU.

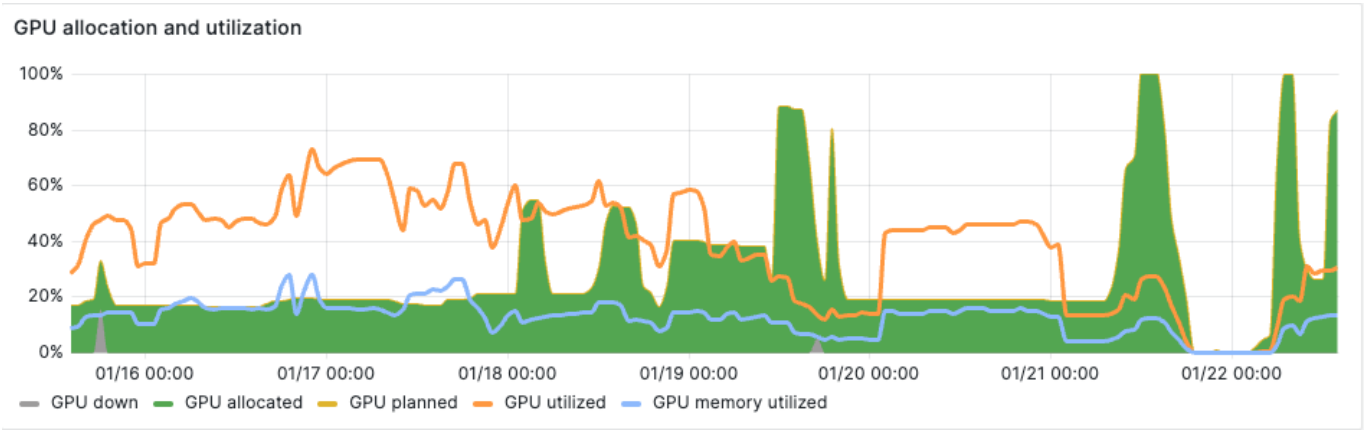


Fig. 1: Accounting data are summarized in the LiSC dashboards. Often, GPU allocation and GPU usage differ.

References

[1] Jumper J, et al. Highly accurate protein structure prediction with AlphaFold. Nature **596**, 583 (2021).

Molecular Dynamic simulations addressing the missing single collision peak in low energy heavy ion scattering

Wolfgang Husinsky **and Richard Wilhelm**

Institut für Angewandte Physik, TU Wien, Austria

The overall goal was to compare experimental and simulation data on the oblique angle scattering of heavy Sn ions at 14 keV energy from a Mo surface. The simulations are performed with the binary collision approximation codes TRIM, TRIDYN, TRI3DYN, SDTrimSP, and IMSIL [1]. Here we present MD simulations, which were performed in the molecular dynamics framework with LAMMPS. A key finding to explain is the absence of an expected peak in the experimental energy spectrum of backscattered Sn ions associated with the pure single collision regime (see Fig. 1). In sharp contrast to this, however, most simulation codes we applied do show a prominent single collision signature both in the energy spectrum and in the angular scatter pattern.

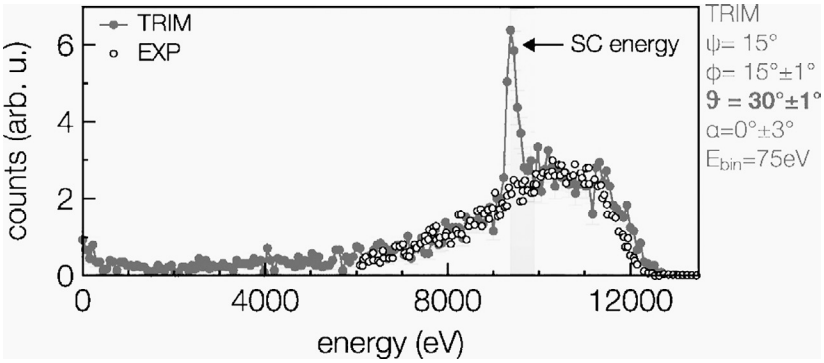


Fig. 1: Problem addressed

Molecular Dynamics (MD) simulations were performed with the LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator) code [2]. The computations were performed on the Vienna Scientific Cluster (VSC). The Mo target consists of approximately 110 000 atoms in the central cell with periodic boundary conditions employed. This size was kept low to make the simulation of one ion trajectory not too computationally expensive, but large enough to keep the damage of the crystal limited to a fraction of the total crystal size and keep the influence on the scattered ions negligible. For the simulations a MEAM/spline (Modified Embedded-Atom Method) potential was used for the Mo atoms and a ZBL potential between the Sn and Mo atoms. The major challenge of the MD simulations of the ion scattering process is to obtain reasonable statistics. To perform a sufficient number of individual simulations we employed a high-performance computing platform of the VSC. SLURM array jobs consisting of 100 members in the job array were set up each using single computing nodes and the associated 48 cores available on them for parallel MD runs on single cores. Under perfectly ideal conditions, this would result in 4800 MD simulations running in parallel. In practice, however, individual nodes will become available only in smaller quantities over time. Hence, the results presented here consumed approximately 100 days of computing time.

References

[1] Wilhelm R.A., Husinsky W., Deuzeman M.J., Rai S. et al, , Nuclear Instruments and Methods in Physics Research B **544**, 165123 (2023).

[2] Thompson A.P. , Aktulga H.M. , et al, LAMMPS - A flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales, Comput. Phys. Comm. **271** 108171 (2022).

NCIS-LA, Nature Computer investigative Simulations of Laser Ablation

Wolfgang Husinsky

Institut für Angewandte Physik, TU Wien, Austria

In our lab we have investigated experimentally over many years the growth of laser induced periodic surface structures (LIPSS), in particular for ultra-short fs laser pulses. The ”nature” and ”physics” of laser ablation - in particular for ultra short laser pulses still poses a lot of puzzles. Thats in spite of the fact of a large number of experiments and also theories. The richness of the physics becomes evident in the understanding of the nature of LIPSS, which have found a wide range of applications. This is due to the small nano-structures one can create below 100 nm. Fig. 1 shows the development of High Frequency Ripples in the 100nm range

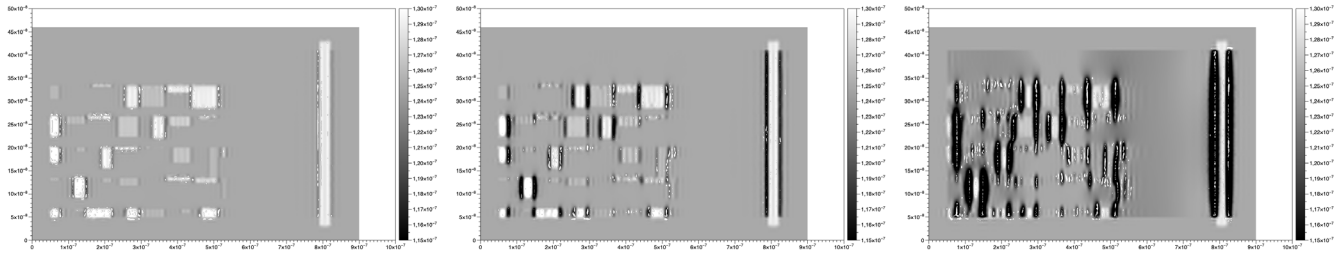


Fig. 1: Development of High Frequency Ripples

The problem for performing such simulations, however, lies in the fact, that several different simulation tools (programs) have to be used and which further more operate at different time- and space- scales. The laser interacts with a material (surface) due to the electric fields it creates in the material. Program tools like COMSOL i.e. use a finite element approach to solve the Maxwell equations and determine the complex field distribution responsible for LIPSS. It is quite evident, that the xyz scale on which theses fields can be calculated is limited. On the other hand the ”response” of the material to these complex fields happens on an atomic scale and can be addressed i.e. by Molecular Dynamic (MD) simulations. The LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator) code is a good choice [1]. In order to simulate reasonable target sizes, the use of an HPC is required. The computations were performed on the Vienna Scientific Cluster (VSC).

Over the last couple of years we have implemented as a core for the simulations COMSOL and LAMMPS. This, however requires a complex data handling and exchange between the programs. We have presented this at an earlier HPC meeting. In the meantime most of the major bottlenecks, which appeared then, have been solved and a simulation of a LIPSS process is now possible within a reasonable time frame. A discussion will be presented. We are convinced, that a major issue in all simulations is the fact, that only a limited number of physical processes can be included and also often in a simplified manner. It will be demonstrated, that specific processes have to be considered in order to achieve the desired results. This might sound trivial, but we can show that Coulomb-like processes, which have often been neglected over the years (especially in simulations), are essential to be included.

References

[1] Thompson A.P. , Aktulga H.M. , et al, LAMMPS - A flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales, Comput. Phys. Comm. **271** 108171 (2022).

Checkpointing on VSC

Dieter Kvasnicka

VSC Research Center, TU Wien, Austria

Checkpointing in the context of high performance computing can be done on several levels, including restarting a loop in a shell script, user programmed checkpointing, using publicly available libraries, and system level checkpointing.

The effort of checkpointing is rewarded with a multitude of benefits: **'Eternal' jobs** overcome the run time limitations (72h) of the queue by checkpointing after e. g. 70 hours. Dumping after initialization and restarting future jobs from the dump **speeds up startup time** by avoiding initialization time. Checkpointing jobs and restarting after an application or system crash improves **fault tolerance**. **Live migration** to a more powerful node (i. e. one with more memory) can save resources. **Cluster maintenance** shortens downtime by checkpointing the cluster before maintenance. Checkpoint jobs in idle queue when other jobs are waiting is used for **priority queueing**. To aid **debugging**, long running jobs are checkpointed just before an error and can be restarted several times from there. Nodes can pause an **interactive session** when the user pauses (e. g. overnight). **Private nodes** can be utilized when idle.

Overall, checkpointing improves the **throughput of the cluster** and reduces the cost per CPU-hour.

Many attempts and implementations have been made in the past decades, some require **kernel patches** or **modifications of user programs**, work with a larger class of programs than others, include open **files** to checkpoints, and vary in many other properties.

Our favorite checkpoint methods are:

DMTCP (Distributed MultiThreaded CheckPointing) runs at the user level, requires (small) modifications of the slurm scripts, is suitable for node-parallel jobs, and has to be started at process start time.

CRIU (Checkpoint/Restore In Userspace) runs at the system level, requires no modifications of user programs, but is restricted to single node jobs.

In the talk differences between methods and their requirements, especially when used at VSC (including Infiniband or Slurm) are highlighted. Various types of applications require different methods.

In many cases the **overhead** of checkpointing is negligible. Affected are the execution time and the storage requirements.

Problems and difficulties when implementing checkpointing on VSC include **kernel versions**, reusage of **process IDs**, local file sytems, the multitude of MPI implementations and the **usage of cgroups by slurm**.

Checkpointing **libraries** which help programmers to implement checkpointing in their programs are also mentioned briefly, since this can be by far the most efficient method.

Since there is no ready-made solution for everybody, methods are compared. An overview of methods of checkpointing are presented to aid the choice of VSC users.

Exploring Energy-Efficient GPU Computing

Markus Hickel[a] **and *Siegfried Höfinger***[a,b]

[a] *VSC Research Center, TU Wien, Austria*
[b] *Department of Physics, Michigan Technological University, USA*

Making use of the roofline model [1] we explore relative shifts in the graph with varying power envelops. Roofline analysis is nowadays routinely available in NVIDIA Nsight Compute (https://docs.nvidia.com/nsight-compute/ProfilingGuide/index.html, see Fig. 1 for an example). We seek maximum levels of practical applicability, hence our target application shall be HPCG (https://www.hpcg-benchmark.org) and variants thereof. Computing platforms shall be NVIDIA's A100 and A40 GPUs (https://developer.nvidia.com/blog/nvidia-ampere-architecture-in-depth) thus from a 1:1 comparison of a particular application one should already be able to unveil the disadvantage of A40 in terms of FP64 performance (1:17). Following environmental as well as economic concerns, energy-efficient high-performance computing is becoming increasingly important in science and research of today. While conventionally reflected in specifically tuned hardware frequently seen in the green500 list (https://www.top500.org/lists/green500/2023/11, see for example the top ranked system "Henri" delivering an impressive 65 GFLOPS/watt), there is also potential for energy-efficient optimizations solely resulting from software enhancements. Particularly interesting in this latter class are power-constraints that set a maximum limit to the power draw of the GPU throughout the execution time of a particular computation. Ideally, such power limits would just decrease the energy consumption without compromising the runtime of the unconstrained (i.e. fully powered) run. To confirm power-limited runtime conditions we plan to utilize the energy-and-power module enabled in VSC's version of LINARO/FORGE/MAP (https://www.linaroforge.com/linaroMap, see Fig. 2 for an example). Another worthwhile extension in this present context is to compare AMD's HIP (https://rocm.docs.amd.com/projects/HIP/en/latest) with NVIDIA's CUDA framework. While the latter has already evolved over a 17-year period, the former is rather recent, but seeks to target GPUs in a more generalized manner, independent of actual architectural specifics.

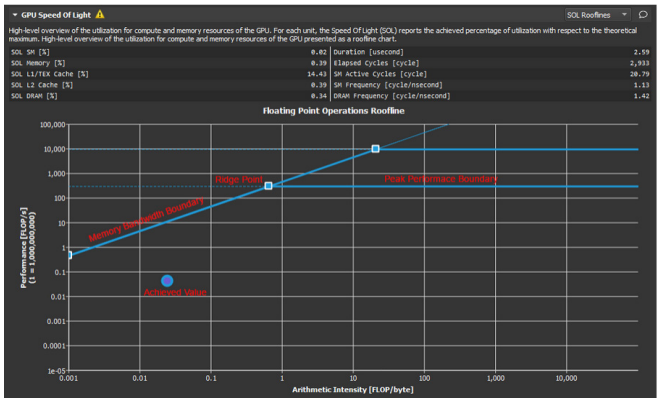


Fig. 1: Example roofline chart available in NVIDIA Nsight Compute (https://docs.nvidia.com/nsight-compute/ProfilingGuide/index.htm).

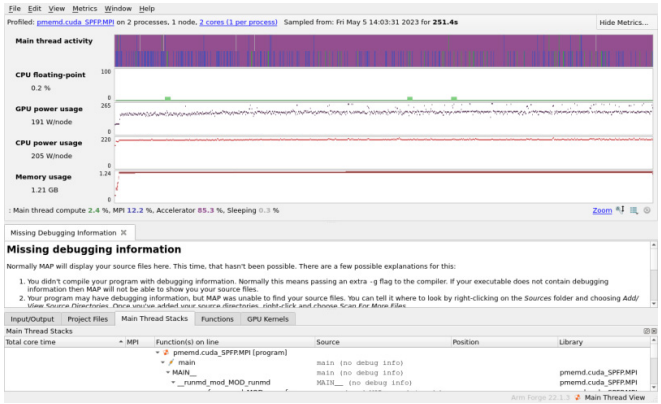


Fig. 2: Example profile involving the energy metrics optionally included in LINARO/FORGE/MAP (https://www.linaroforge.com/linaroMap).

References

[1] Williams, S., Waterman, A., and Patterson, D., Commun. ACM **52(4)**, 65 (2009).

Index of presenting authors

List of ASHPC24 participants

Atzenhofer-Baumgartner, Florian	florian.atzenhofer-baumgartner@uni-graz.at	University of Graz
Benesch, Bettina	bettina.benesch@tuwien.ac.at	University of Vienna and EuroCC Austria
Beranova, Katerina	katerina.beranova@vsb.cz	IT4Innovations, VSB – Technical University of Ostrava
Berger, Sabrina	sabrina.berger@unileoben.ac.at	Montanuniversität Leoben
Bijelic, Marija	marija.bijelic@tuwien.ac.at	University of Vienna and EuroCC Austria
Blaas-Schenner, Claudia	claudia.blaas-schenner@tuwien.ac.at	VSC Research Center, TU Wien and EuroCC Austria
Blaschek, Michael	michael.blaschek@univie.ac.at	University of Vienna
Brandstetter, Dominik	dominik.brandstetter@uni-graz.at	University of Graz
Brank, Matic	matic.brank@fs.uni-lj.si	University of Ljubljana
Brzobohaty, Tomas	tomas.brzobohaty@vsb.cz	IT4Innovations, VSB – Technical University of Ostrava
Coste, Amaury	amaury.coste@ki.si	National Institute of Chemistry, Slovenia
Črnigoj Marc, Tina		Arctur
Deliu, Endri		University of Vienna and EuroCC Austria
Dellago, Christoph	christoph.dellago@univie.ac.at	University of Vienna
Elefante, Stefano	stefano.elefante@ista.ac.at	ISTA – Institute of Science and Technology Austria
Gall, Marian	marian.gall@nscc.sk	Slovak National Supercomputing Centre
Ganguly, Raman	raman.ganguly@univie.ac.at	University of Vienna and EuroCC Austria
Gergely, Eva	eva.gergely@univie.ac.at	University of Vienna and EuroCC Austria
Goiser, Malgorzata	malgorzata.goiser@tuwien.ac.at	VSC Research Center, TU Wien and EuroCC Austria
Goldenberg, Florian	florian.goldenberg@tuwien.ac.at	VSC Research Center, TU Wien
Großöhme, Peter	peter.grossoehme@megware.com	MEGWARE
Gschwandtner, Philipp	philipp.gschwandtner@uibk.ac.at	University of Innsbruck
Guernelli, Moreno	m.guernelli@cineca.it	CINECA
Haase, Gundolf	gundolf.haase@uni-graz.at	University of Graz
Hackl, Benjamin	benjamin.hackl@uni-graz.at	University of Graz
Harisch, Damjan	damjan.harisch@arnes.si	Arnes
Harm, Paul	paul.harm@uni-graz.at	University of Graz
Harrison, Simeon	simeon.harrison@tuwien.ac.at	VSC Research Center, TU Wien and EuroCC Austria
Hartl, Benedikt		TU Wien

Haunschmid, Ernst	ernst.haunschmid@tuwien.ac.at	VSC Research Center, TU Wien
Heistracher, Paul	paul.heistracher@ims.co.at	IMS Nanofabrication GmbH
Hickel, Markus	markus.hickel@tuwien.ac.at	VSC Research Center, TU Wien
Hinsche, Ralph	rhinsche@nvidia.com	NVIDIA
Hirschbrich, Valentin	valentin.hirschbrich@tuwien.ac.at	VSC Research Center, TU Wien
Hoefinger, Siegfried	siegfried.hoefinger@tuwien.ac.at	VSC Research Center, TU Wien
Höller-Lugmayr, Harald	harald.hoeller-lugmayr@ims.co.at	IMS Nanofabrication GmbH
Hunold, Sascha		TU Wien
Husinsky, Wolfgang	husinsky@iap.tuwien.ac.at	TU Wien
Hyryavets, Halyna	halyna.hyryavets@nscc.sk	Slovak National Supercomputing Centre
Javoršek, Jan Jona	jona.javorsek@ijs.si	Jožef Stefan Institute
Jurkovic, Martin	martin.jurkovic@ims.co.at	IMS Nanofabrication GmbH
Karasek, Tomas	tomas.karasek@vsb.cz	IT4Innovations, VSB – Technical University of Ostrava
Kemetmüller, Josef	josef.kemetmueller@ims.co.at	IMS Nanofabrication GmbH
Khalid, Waleed	waleed.khalid@ista.ac.at	ISTA – Institute of Science and Technology Austria
Kirchner, Maria	maria.kirchner@i-med.ac.at	Medical University of Innsbruck
Kozubek, Tomáš	tomas.kozubek@vsb.cz	IT4Innovations, VSB – Technical University of Ostrava
Krasovec, Barbara		Jožef Stefan Institute
Krawczyk-Bryłka, Beata	beabrylk@pg.edu.pl	Gdańsk University of Technology
Kreidl, Sabine	sabine.kreidl@uibk.ac.at	University of Innsbruck
Kriechbaum, Christian	christian.kriechbaum@unileoben.ac.at	Montanuniversität Leoben
Kvasnicka, Dieter	dieter.kvasnicka@tuwien.ac.at	VSC Research Center, TU Wien
Lesjak, Dejan	dejan.lesjak@ijs.si	Jožef Stefan Institute
Lindner, Andreas		Advanced Computing Austria GmbH
Lorenčič, Samo	samo.lorencic@izum.si	IZUM – Institute of information Science, Maribor
Mayerhofer, Thomas	thomas.mayerhofer@inits.at	INiTS and EuroCC Austria
Mazur, Hubert	hubert.mazur@pwr.edu.pl	Wrocław University of Science and Technology
McCartney, Adam	adam.mccartney@tuwien.ac.at	VSC Research Center, TU Wien
McKevitt, James	james.mckevitt@univie.ac.at	University of Vienna
Muck, Katrin	katrin.muck@tuwien.ac.at	VSC Research Center, TU Wien
Neumayer, Michael	michael.neumayer@univie.ac.at	University of Vienna

Nilsson, Jeanette	jeanette.nilsson@ri.se	RISE Research Institutes of Sweden
Oppel, Markus	markus.oppel@univie.ac.at	University of Vienna
Ostermann, Alexander	alexander.ostermann@uibk.ac.at	University of Innsbruck
Pešatová, Karina	karina.pesatova@vsb.cz	IT4Innovations, VSB – Technical University of Ostrava
Pfahler, David	david.pfahler@ims.co.at	IMS Nanofabrication GmbH
Pfennig, Tobias	tobias.pfennig@megware.com	MEGWARE
Pfister, Martin	martin.pfister@tuwien.ac.at	University of Vienna and EuroCC Austria
Pilipović, Ratko	ratko.pilipovic@fri.uni-lj.si	University of Ljubljana
Pitonak, Michal	michal.pitonak@nscc.sk	Slovak National Supercomputing Centre
Povh, Janez	janez.povh@rudolfovo.eu	Rudolfovo – Science and technology center Novo mesto and University of Ljubljana
Praprotnik, Matej	praprot@cmm.ki.si	National Institute of Chemistry, Slovenia
Prica, Teo	teo.prica@izum.si	IZUM – Institute of information Science, Maribor
Pukšič, Boris	boris.puksic1@um.si	Knowledge and Technology Transfer Office, University of Maribor
Rattei, Thomas	thomas.rattei@univie.ac.at	University of Vienna
Reiter, Eduard	eduard.reiter@uibk.ac.at	University of Innsbruck
Remizova, Anna	anna.remizova@tuwien.ac.at	University of Vienna and EuroCC Austria
Riha, Lubomir	lubomir.riha@vsb.cz	IT4Innovations, VSB – Technical University of Ostrava
Robnik-Šikonja, Marko	marko.robnik@fri.uni-lj.si	University of Ljubljana
Rosina, Johannes	johannes.rosina@jku.at	Johannes Kepler University Linz
Sangiogo Gil, Eduarda	eduarda.sangiogo.gil@univie.ac.at	University of Vienna
Sattari, Sanaz	sanaz.sattari@tuwien.ac.at	VSC Research Center, TU Wien
Schicker, Irene	irene.schicker@geosphere.at	GeoSphere Austria
Schlögl, Alois	alois.schloegl@ist.ac.at	ISTA – Institute of Science and Technology Austria
Schmidt, Thorsten		Cornelis Networks Inc.
Schnepf, Bastian	schnepf@altair.com	Altair
Schuller, Lukas	lukas.schuller@ims.co.at	IMS Nanofabrication GmbH
Schulze, Hendrik	hendrik.schulze@ims.co.at	IMS Nanofabrication GmbH
Siddique, Kashif	kashif.siddique@uibk.ac.at	University of Innsbruck
Siegel, Moritz	moritz.siegel@tuwien.ac.at	VSC Research Center, TU Wien
Šikaleska, Violeta	violeta.sikaleska@izum.si	IZUM – Institute of information Science, Maribor

Sinai, Gregor	gregor.sinai@inits.at	INiTS
Slejko, Ema	ema.slejko@ki.si	National Institute of Chemistry, Slovenia and University of Ljubljana
Sluga, Davor	davor.sluga@fri.uni-lj.si	University of Ljubljana
Špoljar, Jurica	jspoljar@srce.hr	University of Zagreb, University Computing Centre (SRCE)
Štepec, Dejan	dejan.stepec@patherea.ai	University of Ljubljana
Stöhr, Markus	markus.stoehr@boku.ac.at	University of Natural Resources and Life Sciences, Vienna and EuroCC Austria
Störi, Herbert	herbert.stoeri@tuwien.ac.at	VSC Research Center, TU Wien
Surovovs, Kirils	kirils.surovovs@lu.lv	University of Latvia
ter Hoeven, Roeland	r.hoeven@parityqc.com	University of Innsbruck and Parity Quantum Computing GmbH
Terjék, Mihály	terjek.mihaly@kifu.gov.hu	KIFÜ – Governmental Agency for IT Development
Thaler, Martin	martin.thaler@uibk.ac.at	University of Innsbruck
Thalhammer, Mechthild	mechthild.thalhammer@uibk.ac.at	University of Innsbruck
Thoman, Peter	peter.thoman@uibk.ac.at	University of Innsbruck
Tomic, Drasko	drasko@irb.hr	Ruđer Bošković Institute, Croatia
Tomšič, Pavel	pavel.tomsic@fs.uni-lj.si	University of Ljubljana
Trzepizur, Natascha	natascha.trzepizur@inits.at	INiTS and EuroCC Austria
Tykierko, Mateusz	mateusz.tykierko@pwr.edu.pl	Wroclaw Centre for Networking and Supercomputing
Ujlaki, Gyula	ujlaki.gyula@kifu.gov.hu	KIFÜ – Governmental Agency for IT Development
Valh, Dejan	dejan.valh@izum.si	IZUM – Institute of information Science, Maribor
Vardas, Ioannis		TU Wien
Vasileska, Ivona	ivona.vasileska@fs.uni-lj.si	University of Ljubljana
Vialov, Ivan	ivan.vialov@tuwien.ac.at	VSC Research Center, TU Wien
Vianello, Marco	marcov@math.unipd.it	Dept. of Mathematics, University of Padova
Vodopivec, Urša	ursa.vodopivec@arnes.si	Arnes
Voigt, Aiko	aiko.voigt@univie.ac.at	University of Vienna
Wang, Yin	yin.wang@uibk.ac.at	University of Innsbruck
Wimmer, Elias	elias.wimmer@tuwien.ac.at	VSC Research Center, TU Wien
Wimmer, Iuliia	julia.wimmer@tuwien.ac.at	VSC Research Center, TU Wien and University of Vienna and EuroCC Austria
Zabloudil, Jan	jan.zabloudil@tuwien.ac.at	University of Natural Resources and Life Sciences, Vienna

Zamuda, Aleš	ales.zamuda@um.si	University of Maribor
Zauchner, Mario	mario.zauchner@ims.co.at	IMS Nanofabrication GmbH
Zebec, Ziga	ziga.zebec@izum.si	IZUM – Institute of information Science, Maribor

Fourteen participants preferred not to be listed.

DOI: https://doi.org/10.25365/phaidra.463

ISBN: 978-3-200-09645-5

Published by:
EuroCC Austria
c/o Universität Wien
Universitätsring 1
1010 Vienna, Austria
https://eurocc-austria.at/

Edited by:
Eduard Reiter, Research Area Scientific Computing, University of Innsbruck, Austria

Layout:
Irene Reichl and Claudia Blaas-Schenner, VSC Research Center, TU Wien, 2016