dealii-X: Preparing generic PDE solvers for exascale supercomputers

Martin Kronbichler Pasqua D'Ambra Patrick Amestoy Alfredo Buttari Fabio Durastante Marco Feder Salvatore Filippone Jean-Yves L'Excellent Luca Heltai Theo Mary Chiara Puglisi

Ruhr University Bochum, Germany

In collaboration with the dealii-X CoE and the deal.II community

SPONSORED BY THE Federal Ministry of Education and Research

Funded by the European Union NextGenerationEU

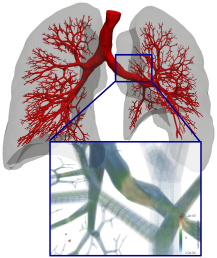
EuroHPC Joint Undertaking

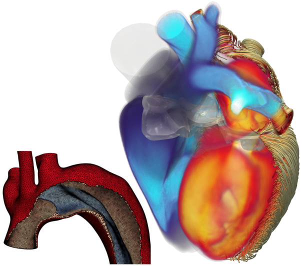
June 13, 2025

dealii-X: an Exascale Framework for Digital Twins of the Human Body

RUHR UNIVERSITÄT BOCHUM RUB

- A scalable, high-performance computational platform
- Using the deal.II finite element library
- Create accurate digital twins of human organs
- Enable exascale computations of complex PDE models with deal.II

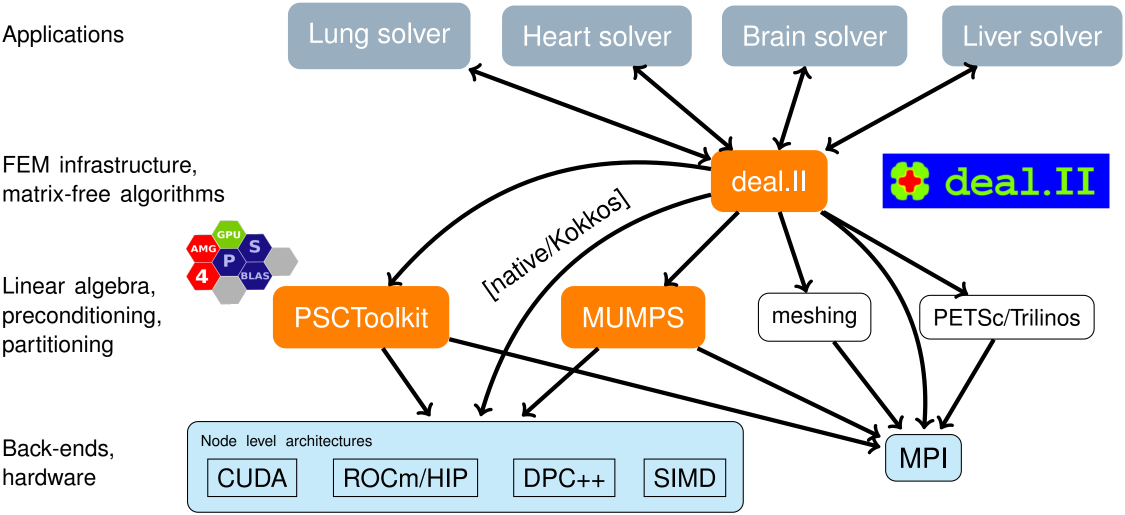




Project partners

- Ruhr University Bochum, DE
- Università di Pisa, IT
- Università degli Studi di Brescia, IT
- Leibniz Supercomputing Centre (LRZ), DE
- Scuola Internazionale Superiore degli Studi Avanzati, IT
- Università degli Studi di Roma Tor Vergata , IT
  - CNR
- Institut National Polytechnique de Toulouse, FR
  - Centre National de la Recherche Scientifique
- Technical University of Munich, DE
- Politecnico di Milano, IT
- Friedrich-Alexander University Erlangen-Nuremberg, DE
- Forschungsverbund Berlin eV, DE
- Exact Lab SRL, IT
  - Dualistic
- Virtual Physiological Human Institute, BE

dealii-X: Composition of software libraries



Core software project: deal.II

- deal.II: a **finite element library**
  - Provide the building blocks to easily build a solver
  - Applicable to almost any partial differential equation: Make it easy to state weak form in finite element problem
- Functionality for mathematical ingredients in a finite element problem
- Write basic prototypes in 100–200 lines of code or large applications with $>$ 100k LOC
- Pre- and post-processing with most formats
- *h*, *p* and *hp* adaptive methods
- Linear algebra in deal.II or via external packages (LAPACK, PETSc, Trilinos, Ginkgo, PSCToolkit, MUMPS, SUNDIALS, . . .)
- Parallelization with MPI, threads, GPUs with Kokkos, . . .
- 2025 SIAM/ACM Prize in Computational Science and Engineering

Arndt, Bangerth, Davydov, Heister, Heltai, Kronbichler, Maier, Pelteret, Turcksin and Wells: The deal.II finite element library: design, features, and insights. *Computers & Mathematics with Applications* 81:407–422, 2021 `doi:10.1016/j.camwa.2020.02.022`

Central algorithm in deal.II: matrix-free operator evaluation

Matrix-vector product

Motivation: sparse linear algebra mostly limited by memory bandwidth, **re-compute rather than load** to get **speedup**

matrix-based:

$$\begin{cases} A = \sum_{e=1}^{N_{\rm el}} P_e^{\rm T} A_e P_e & \text{(assembly)} \\ v = Au & \text{(matrix-vector product within iterative solver)} \end{cases}$$

↓

matrix-free:

$$v = \sum_{e=1}^{N_{\rm el}} P_e^{\rm T} A_e \left(P_e u\right)$$

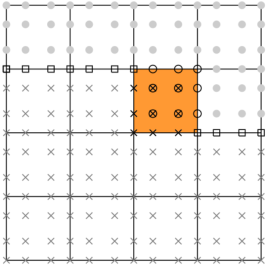
Matrix-free evaluation of FEM operator:

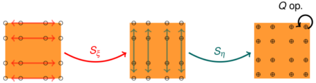
Loop over elements $e = 1, \ldots, N_{\rm el}$:

(i) Extract local vector values: $u_e = P_e u$

(ii) Apply operation locally by integration: $v_e = A_e u_e$, **do not form** A_e, compute its action by FEM integrals

(iii) Sum results from (ii) into the global solution vector: $v = v + P_e^{\rm T} v_e$





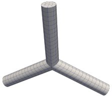
$$A_e u_e = S^{\rm T} Q_e S u_e$$

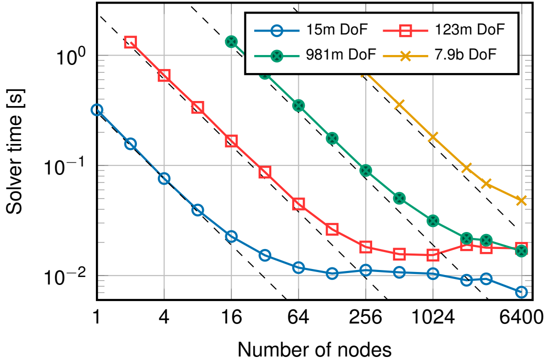
M. Kronbichler, K. Kormann, A generic interface for parallel finite element operator application. *Comput. Fluids* 63:135–147, 2012
M. Kronbichler, K. Kormann, Fast matrix-free evaluation of discontinuous Galerkin finite element operators. *ACM TOMS* 45(3), 29, 2019

Scalability of multigrid solver on SuperMUC-NG (Intel Skylake CPU)

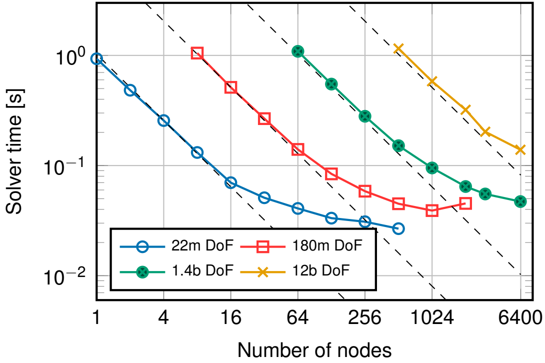
Point-Jacobi/Chebyshev (3,3) smoother, tolerance 10^{-3}

- Generic bifurcation
- Converges in 3 CG iterations





- Lung geometry
- Converges in 7 CG iterations



[1] Kronbichler, Fehn, Munch, Bergbauer, Wichmann, Geitner, Allalen, Schulz, Wall: A next-generation discontinuous Galerkin fluid dynamics solver with application to high-resolution lung airflow simulations. In *SC'21: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, (2021).

Parallel Sparse Computation Toolkit – `psctoolkit.github.io`

Two central libraries PSBLAS and AMG4PSBLAS:

- Existing software standards:
  - MPI, OpenMP, CUDA
  - Serial sparse BLAS,
  - (Par)Metis,
  - AMD
- Attention to performance using **modern Fortran**;
- Research on new preconditioners;
- No need to delve in the data structures for the user;
- Tools for error and mesh handling beyond simple algebraic operations;
- **D**istributed **Sparse BLAS**;
- Standard **K**rylov **s**olvers: CG, FCG, (R)GMRES, BiCGStab, CGS, . . .

Two central libraries PSBLAS and AMG4PSBLAS:

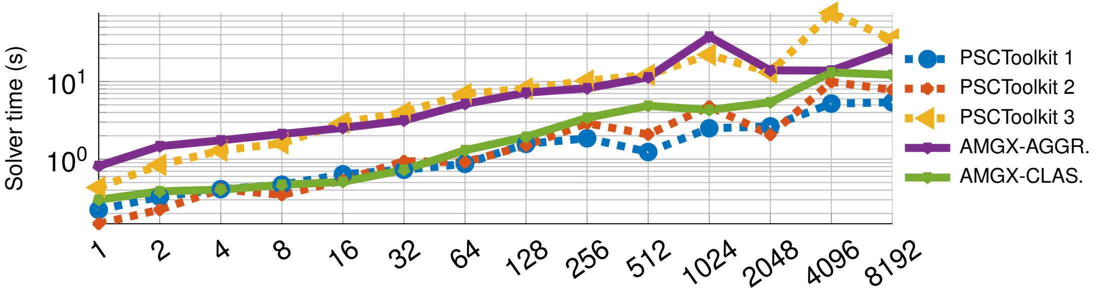
- Domain decomposition preconditioners
- **A**lgebraic **M**ulti**G**rid with aggregation schemes
  - Vaněk, Mandel, Brezina Aggregation
  - Matching Based
  - Smoothed Aggregation
- Parallel Smoothers (Block-Jacobi, Hybrid-GS/SGS/FBGS, ℓ_1 variants) that can be coupled with specialized block (approximate) solvers MUMPS, SuperLU, Incomplete Factorizations (AINV, INVK/L, ILU-type), and with Polynomial Accelerators (Chebyshev 1^{st}-kind, Chebyshev 4^{th}-kind)
- V-Cycle, W-Cycle, K-Cycle

Parallel Sparse Computation Toolkit – `psctoolkit.github.io`

Two central libraries PSBLAS and AMG4PSBLAS.

- Freely available from: `https://psctoolkit.github.io`,
- Open Source with BSD 3 Clause License,
- Can be compiled/installed with either *Automake*/*CMake* or *Spack.io*: “`spack install psblas`”.
- See: D’Ambra, P., F. Durastante, and S. Filippone. “Parallel Sparse Computation Toolkit.” *Software Impacts* 15 (2023): 100463; for a **description of the architecture**.

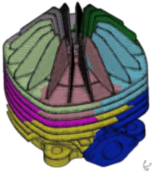
3D Poisson benchmark: PSCToolkit vs NVIDIA AMGx



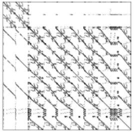
- *Weak scaling* with 8×10^6 unknowns per GPU, 1 to 8192 A100 GPUs on the **Leonardo Supercomputer**.
- Full details in preprint[2]

[2] D'Ambra, P., Durastante, F. & Filippone, S. PSCToolkit: solving sparse linear systems with a large number of GPUs. (2025), arXiv 2406.19754

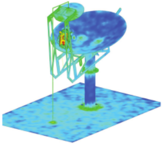
Sparse direct solvers



Code Aster (EDF)



Wide range of applications (e.g. medical diagnostics, geoscience, electromagnetism, circuit simulation, structural analysis . . .)



FEKO-EM (Altair)

Sparse direct linear solvers

Factor $\mathbf{A} = \mathbf{LU}$; Solve: $\mathbf{LY} = \mathbf{B}$, then $\mathbf{UX} = \mathbf{Y}$

Method of choice for its accuracy and robustness

MUMPS: a MUltifrontal Massively Parallel Solver

MUMPS

A robust package using a direct method for solving

$$\mathbf{A}\mathbf{X} = \mathbf{B},$$

where $\mathbf{A}$ is a large sparse matrix, and $\mathbf{X}, \mathbf{B}$ are dense or sparse

A free software distributed under CeCILL-C license (LGPL like), co-developped by Bordeaux Univ., CERFACS, CNRS, ENS Lyon, INPT, Inria, Mumps Tech, and Sorbonne Univ.

MUMPS Solver has been awarded by the European Mathematical Society (EMS) and the European Consortium for Mathematics in Industry (ECMI), the Lanczos Prize for Mathematical Software

In this talk: Recent work on data sparsity and mixed precision on large scale applications

- Data sparsity relies on Block Low-Rank (BLR) compression to compute an approximate factorization $\mathbf{A} \approx \mathbf{L}_\varepsilon \mathbf{U}_\varepsilon$ at accuracy ε controlled by the user[3]
- BLR reduces asymptotic complexity: (3D Poisson, $n = N \times N \times N$ mesh):[4]

$$\mathcal{O}(n^2) \rightarrow \mathcal{O}(n^{4/3}) \text{ flops}$$
$$\mathcal{O}(n^{4/3}) \rightarrow \mathcal{O}(n \log n) \text{ memory}$$

- BLR has been proved to be numerically stable[5]
- BLR is a favorable playground to introduce mixed precision for computation and storage[6]

[3] P. Amestoy, C. Ashcraft, O. Boiteau, A. Buttari, J.-Y. L'Excellent, and C. Weisbecker. "Improving Multifrontal Methods by Means of Block Low-Rank Representations". *SIAM SISC* (2015).

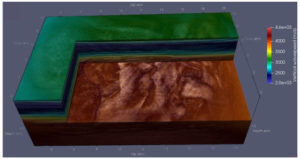
[4] P. Amestoy, A. Buttari, J.-Y. L'Excellent, and T. Mary. "On the Complexity of the Block Low-Rank Multifrontal Factorization". *SIAM SISC* (2017).

[5] N. Higham and T. Mary. "Solving Block Low-Rank Linear Systems by LU Factorization is Numerically Stable". *IMA J. Numer. Anal.* (2021).

[6] Amestoy, Boiteau, Buttari, Gerest, Jézéquel, L'Excellent, Mary. "Mixed precision low-rank approximations and their application to BLR LU factorization". *IMA J. Numer. Anal.* (2023).

BLR with Mixed Precision on Full-Waveform Inversion

- Adastra computer (CINES national computing center) (CPU partition based on AMD GENOA nodes with 192 cores, AMD EPYC 9654, 2.4GHz)
- Application: Gorgon Model, reservoir 23km $\times$ 11km $\times$ 6.5km, Helmholtz equation $\Rightarrow$ Complex matrix, 531 Million dofs
  - Flops for one Full-Rank LU factorization= 2.6×10^{18}
  - Full Waveform Inversion: 30 Millions CPU hours



(25-Hz Gorgon FWI velocity model)

Example simulation

FR (Full Rank, fp32); BLR with $\varepsilon_{\text{blr}} = 10^{-5}$, mixed-precision BLR with 3 precisions (32 bits, 24 bits, 16 bits) for storage

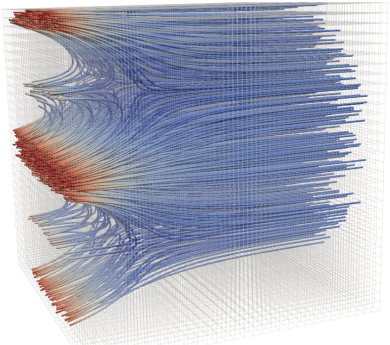
LU size (TBytes)			Flops		Time BLR + Mixed (sec)			Scaled Resid.
FR	BLR	+mixed	FR	BLR+mixed	Analysis	Facto	Solve	BLR+mixed
73	34	26	$2.6 \cdot 10^{18}$	0.5×10^{18}	334	5500	27	7×10^{-4}

In practice: hundreds to thousands of Solve steps (sparse right-hand sides (sources))

[7] Operto et al. “Pushing the limits of 3D frequency-domain FWI with the 2015/2016 OBN Gorgon dataset”. *EAGE 2024 conference* (2024).

MUMPS robustness for 3D Stokes problem

RUHR UNIVERSITÄT BOCHUM RUB



Streamlines for velocity field **u**

- Kovasznay problem with $\mathcal{Q}_2\mathcal{Q}_1$ elements
- GMRES + BLR preconditioner for whole system
- Low-rank threshold: $\varepsilon = 10^{-6}$
- Compare with MinRes + diagonal preconditioner
- AMG V-cycle for velocity block, CG + pressure mass matrix for Schur complement
- Report iteration counts and times (setup + solve phase) [s]

Comparison GMRES vs. MinRes, 16 cores

# dofs ($\mathbf{u} + p$)	MUMPS + GMRES	MinRes
15,468	3 (8.1e-1)	72 (1.15e0)
112,724	5 (5.2e0)	73 (1.89e0)
859,812	5 (6.6e1)	75 (1.1e1)
6,714,692	6 (1.2e3)	77 (1.2e2)

Summary

- dealii-X is a new CoE started in Oct 2024
- Focus on robust solvers for coupled multi-physics problems
- Work on iterative solvers with matrix-free ingredients (very fast), sparse iterative solvers with robust preconditioners via PSCToolkit (fast, medium robust) and sparse direct solvers with MUMPS (slower, robust)
- Mixed precision, GPU portability, scalability to large node counts work in progress