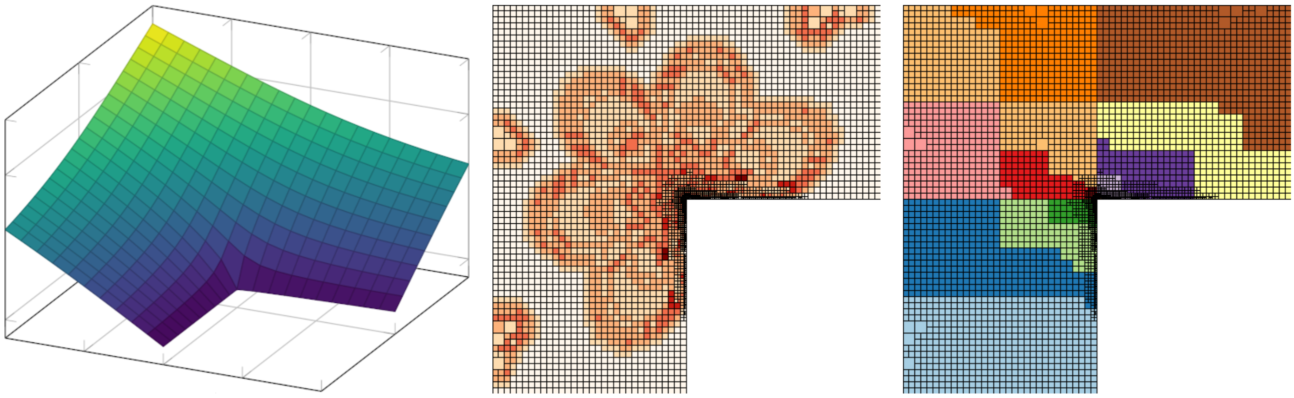


ALGORITHMS FOR MASSIVELY PARALLEL GENERIC HP-ADAPTIVE FEM

June 05, 2020 | Marc Fehling | m.fehling@fz-juelich.de |

BERGISCHE UNIVERSITÄT WUPPERTAL

JÜLICH Forschungszentrum

Table of contents

Fire safety science

- Civilian safety in metro stations
  - Smoke spread in case of fire
  - Egress routes for pedestrians

- Individual examinations necessary on complex geometries
- Experiments expensive $\longrightarrow$ Alternative: Computer simulations

Figure: Experiments in metro

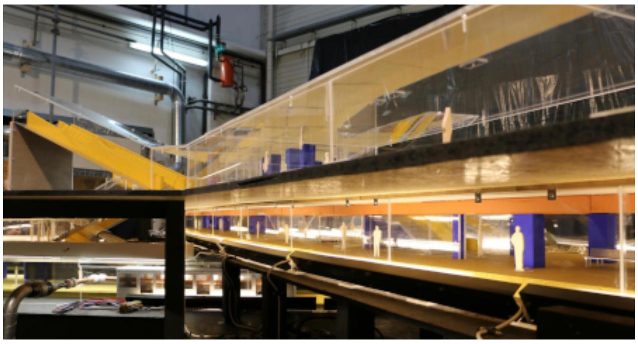


Figure: Physical model (scale 1:15)

Fire simulation

- Lots of software tools available: FDS, FireFOAM, Ansys Fluent, ...
- Flexible: Scenarios may be varied easily

- **But:** Large and complex geometries yield lots of workload
- Simulations require a lot of time

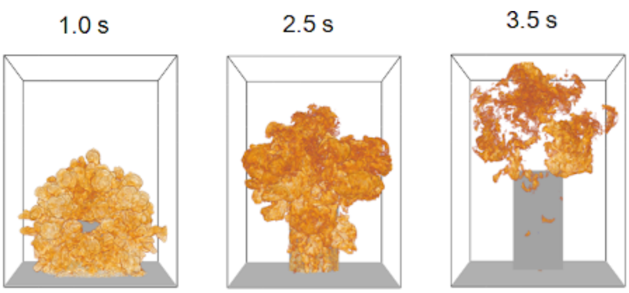


Figure: Deflagration of Heptane

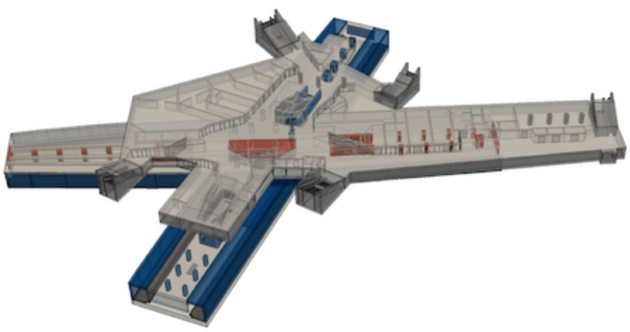


Figure: CAD model

Computational fluid dynamics

- Smoke spread modeled with incompressible Navier–Stokes equations

$$\nabla \cdot \boldsymbol{u} = 0$$
$$\rho_0 \left[\partial_t \boldsymbol{u} + (\boldsymbol{u} \cdot \nabla)\, \boldsymbol{u}\right] = -\nabla p + \nabla \cdot (2\, \mu\, \boldsymbol{\epsilon}) + \boldsymbol{f}$$
$$\rho_0 c_p \left[\partial_t T + (\boldsymbol{u} \cdot \nabla)\, T\right] = 2\, \mu\, \boldsymbol{\epsilon} : \nabla \boldsymbol{u} + \nabla \cdot (\kappa\, \nabla T) + q - \boldsymbol{u} \cdot \boldsymbol{f}$$

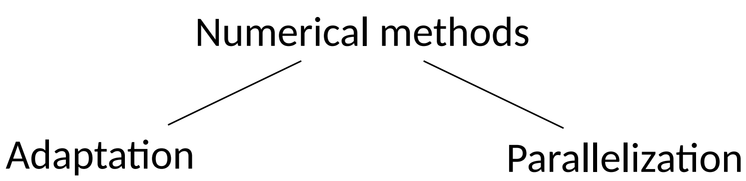
- Solution via numerical methods $\longrightarrow$ Computational fluid dynamics
- High resolution necessary for ...
  - Large gradients in temperature and velocity
  - Turbulence
  - Flow separation

Motivation for PhD project

- Fires stay localized in general, not only during ignition phase
- Unnecessarily fine grids bind resources that could be used near the fire
- Demand for effective use of computing power

Goal

- Balance accuracy and workload by adapting resolution
- Accelerate simulations by exploiting hardware



Example: Adaptive mesh refinement

- Demonstration of adaptive mesh refinement (*h*-adaptive methods) via moving vortex test case as a shape-preserving potential stream.

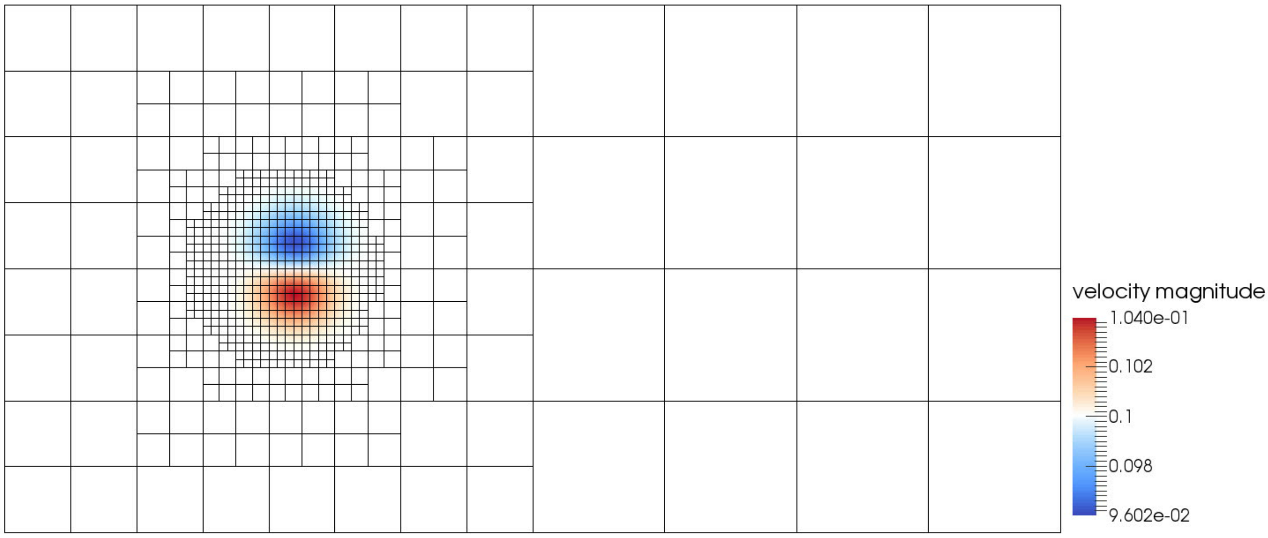


Figure: Video of velocity magnitude of moving vortex, overlaid with current mesh

Numerical methods

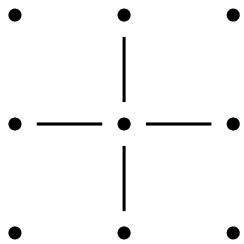
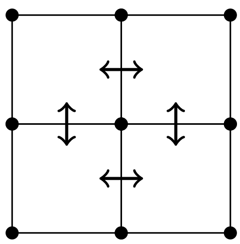
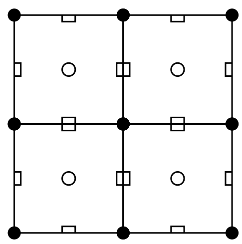
Finite differences	Finite volumes	Finite elements
		
Difference quotients as differential operators	Balance fluxes on faces between volumes Conservation laws	Limit function space to piecewise polynomials
h-adaptive methods	*h*-adaptive methods	*hp*-adaptive methods

Table of contents

Introduction

Fire safety science

Computation fluid dynamics

hp-adaptive methods

Finite Element Method

Adaptive methods

Example: Laplace equation

Parallelization

Example: Laplace equation

Summary & Outlook

Finite element method

- Shape functions form nodal basis

$$\varphi_i(x_j) = \delta_{ij} = \begin{cases} 1 & \text{for } i = j \\ 0 & \text{for } i \neq j \end{cases}$$

- Q_p elements from Lagrange interpolation with degree p

- Finite element approximation is linear combination of shape functions

$$u_{hp}(x) = \sum_i u_i \, \varphi_i(x)$$

- Coefficients u_i are degrees of freedom

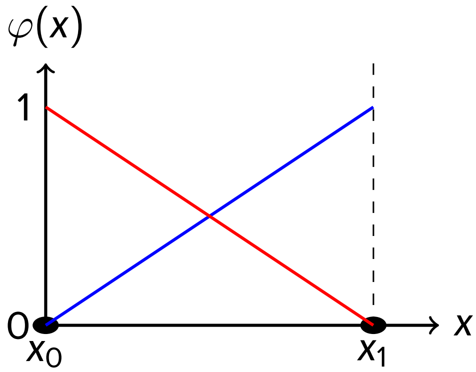


Figure: Q_1 element

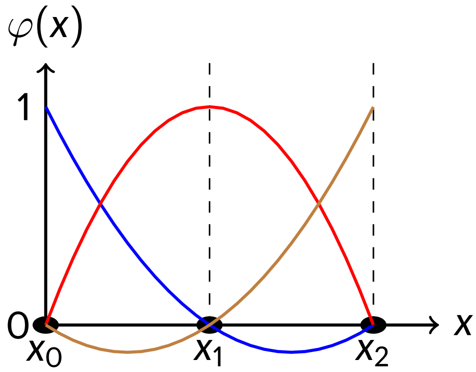


Figure: Q_2 element

Adaptive methods

- Focus computational resources on areas of interest
- Align simulation resolution with complexity of current solution

- Finite Element Method (FEM) provides two different possibilities:

***h*-adaptation:**	dynamic cell sizes	good for irregular solutions
***p*-adaptation:**	dynamic function spaces	good for smooth solutions

- Combination of both possible

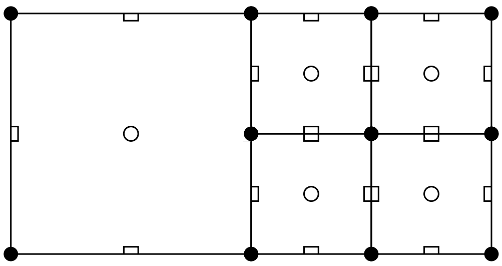


Figure: *h*-adaptivity

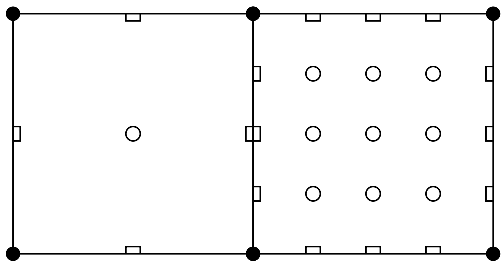


Figure: *p*-adaptivity

Adaptation criteria

- **Which** cells to adapt?
- **How** to adapt? *h*/*p*?

- Manual adaptation
- Automatic adaptation
  - Criterion to indicate adaptation
  - General approach -OR- tied to the problem

- Automatic *hp*-decision strategies discussed in the dissertation
  1. Error prediction based on refinement history [Melenk and Wohlmuth, 2001]
  2. Smoothness estimation by decay of Fourier coefficients [Bangerth and Kayser-Herold, 2009]
  3. Smoothness estimation by decay of Legendre coefficients [Mavriplis, 1994]

Example: Reentrant corner

- Singularity at reentrant corners for elliptic problems
- L-shaped domain:

$$\Omega = [-1, 1]^2 \setminus ([0, 1] \times [-1, 0])$$

- Manufactured Laplace problem

$$\begin{aligned} -\nabla^2 u &= 0 \qquad \text{on} \quad \Omega \\ u &= \bar{u} \qquad \text{on} \quad \partial\Omega \end{aligned}$$

$$\bar{u} = r^{2/3} \sin(2/3\,\varphi)$$

$$\|\nabla \bar{u}\| = r^{-1/3}$$

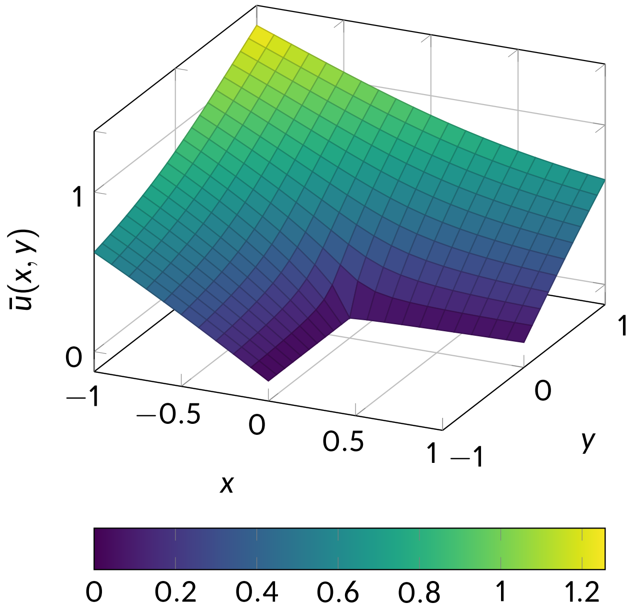


Figure: L-shaped domain

Example: Successive refinement

- Initialize coarse mesh
- Solve and refine in multiple cycles for tailored discretization

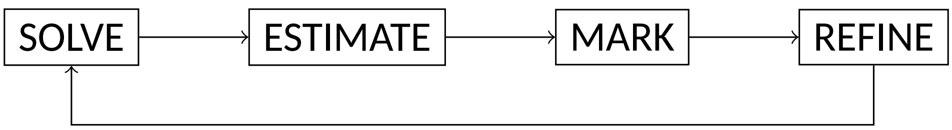


Figure: Successive refinement

1. Calculate refinement criteria (here: error estimates)
2. Flag 30%/3% of cells with highest/lowest criterion for refinement/coarsening
3. Calculate decision criteria (here: smoothness estimates)
4. Flag 90%/10% for *p*-/*h*-adaptation

Example: Successive refinement

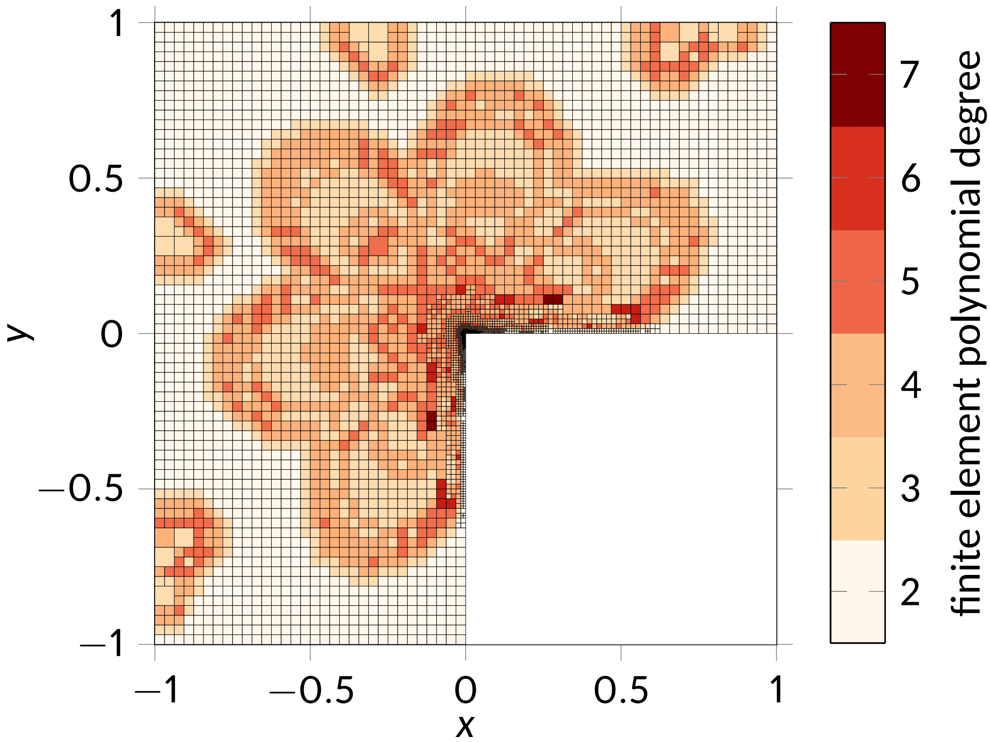
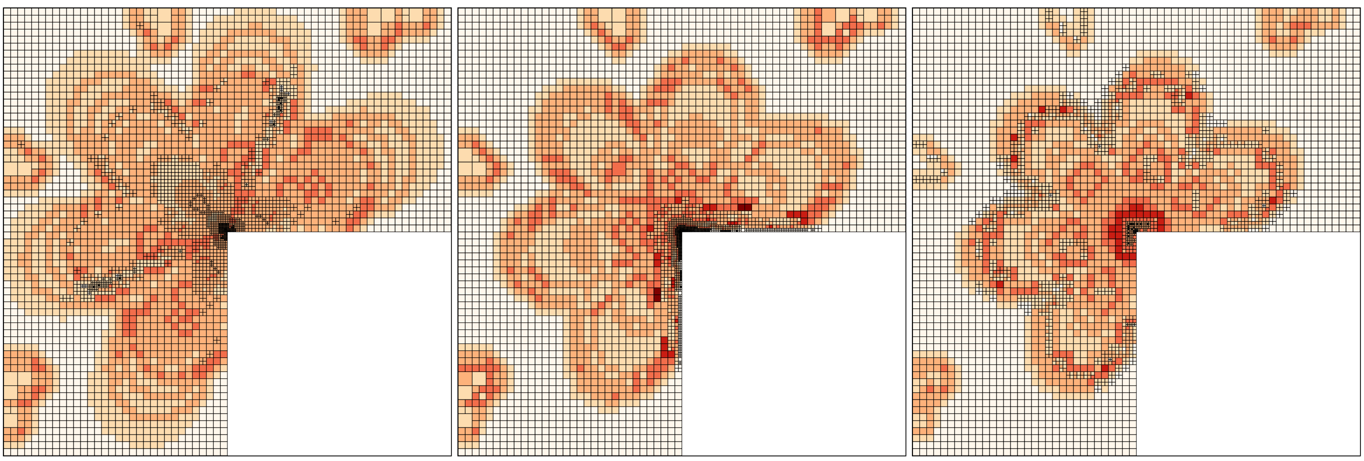


Figure: Polynomial degrees in cycle 5. Zoom 100%.

Example: Successive refinement



(a) Fourier coefficient decay (b) Legendre coefficient decay (c) Refinement history

Figure: Mesh and polynomial degrees of finite elements after 5 consecutive *hp*-adaptations.

Example: Successive refinement

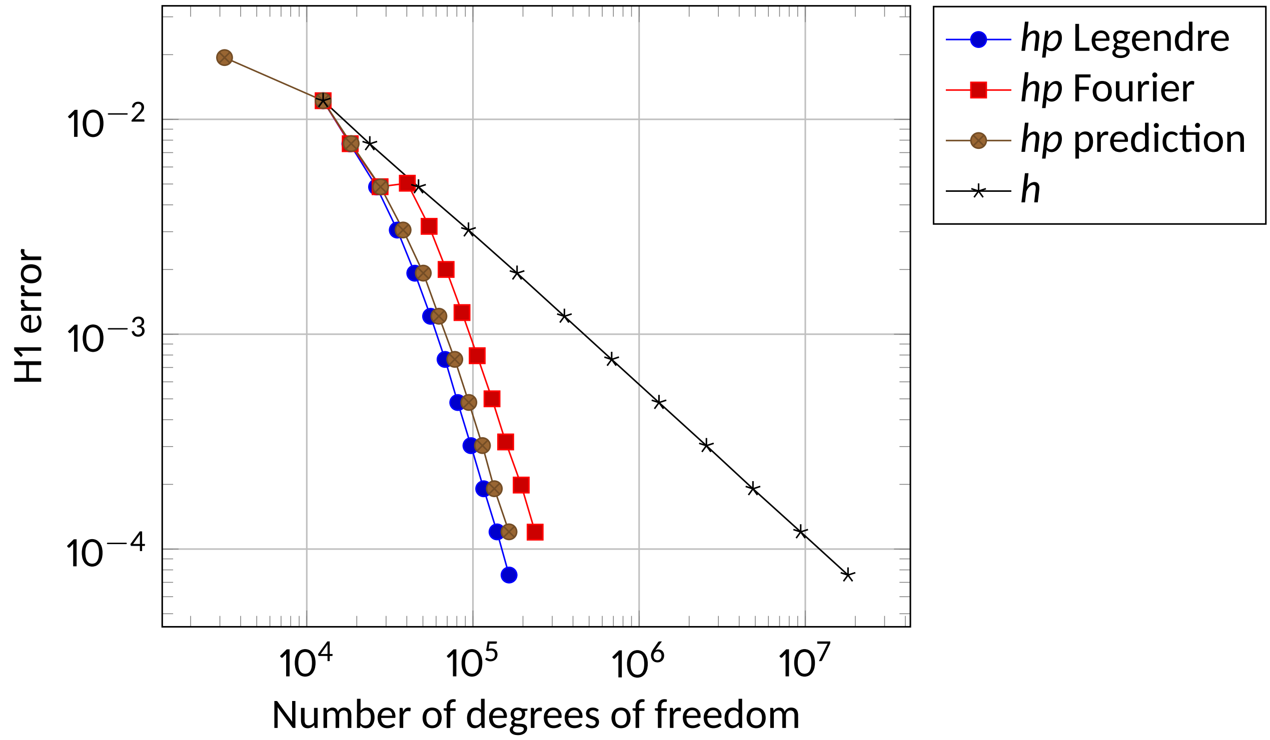
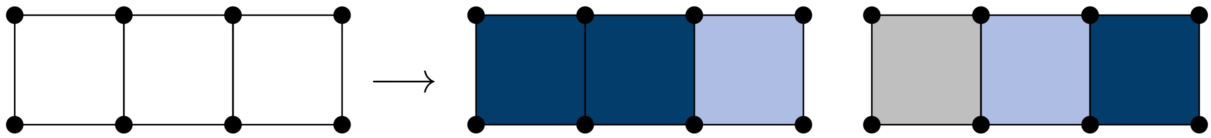


Figure: Error convergence for different strategies

Table of contents

Parallelization

- Current computer architectures provide multi-core processors
  - Supercomputers arrange those on distributed nodes
- Using all resources efficiently requires parallelization
  - Distribution of workload and memory demand
- Our approach: Distribution of domain on several processes
  - Each subdomain needs relevant part of the global solution
  - Requires a layer of so called ghost cells
  - Involves communication between processors

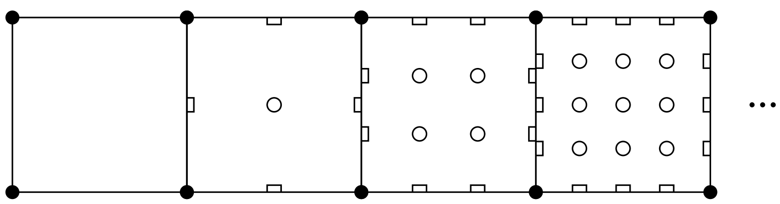


(a) Domain to be distributed (b) Subdomain for CPU 0 (c) Subdomain for CPU 1

Figure: Illustration of **locally owned**, ghost, and artificial cells

Parallel hp-adaptive FEM

- Combination of hp-adaptive methods with parallelisation
- The non-trivial parts are:
  1. Enumeration of degrees of freedom, independent of number of subdomains
  2. Consignment of contiguous memory chunks for data transfer
  3. Weighted repartitioning for load balancing



Q_1	Q_2	Q_3	Q_4
4 nodes	9 nodes	16 nodes	25 nodes

Figure: Different finite elements and their number of nodes in 2D

Example: Load balancing

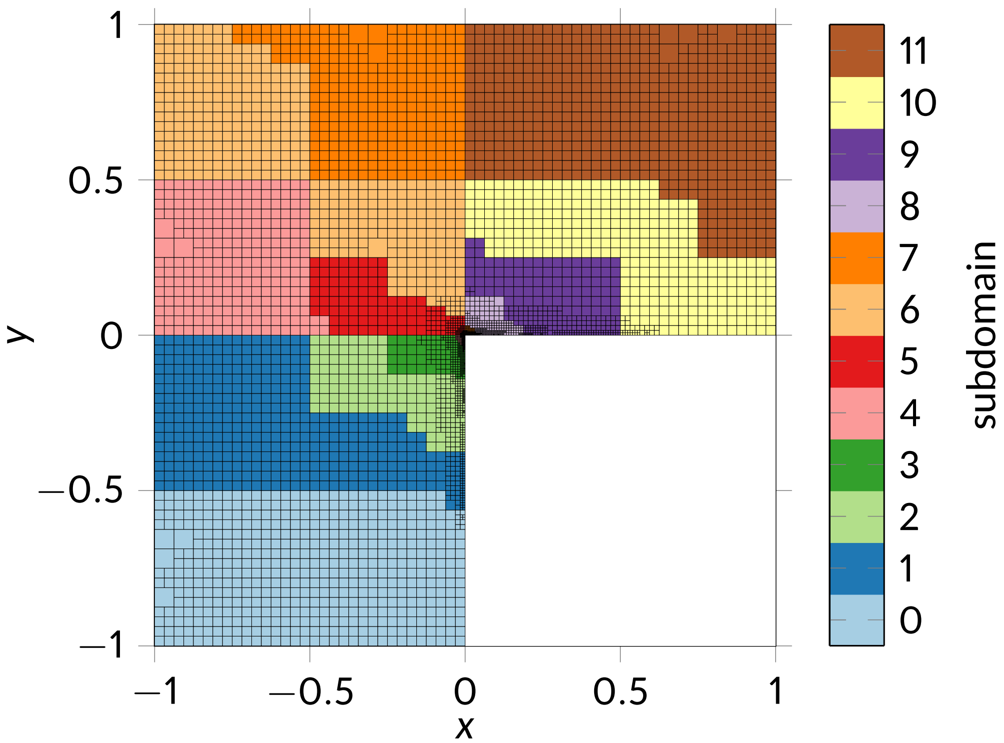


Figure: Mesh decomposition in cycle 5. Weights assigned to cells are $\propto n_{\text{dofs}}^{1.9}$.

Example: Strong scaling

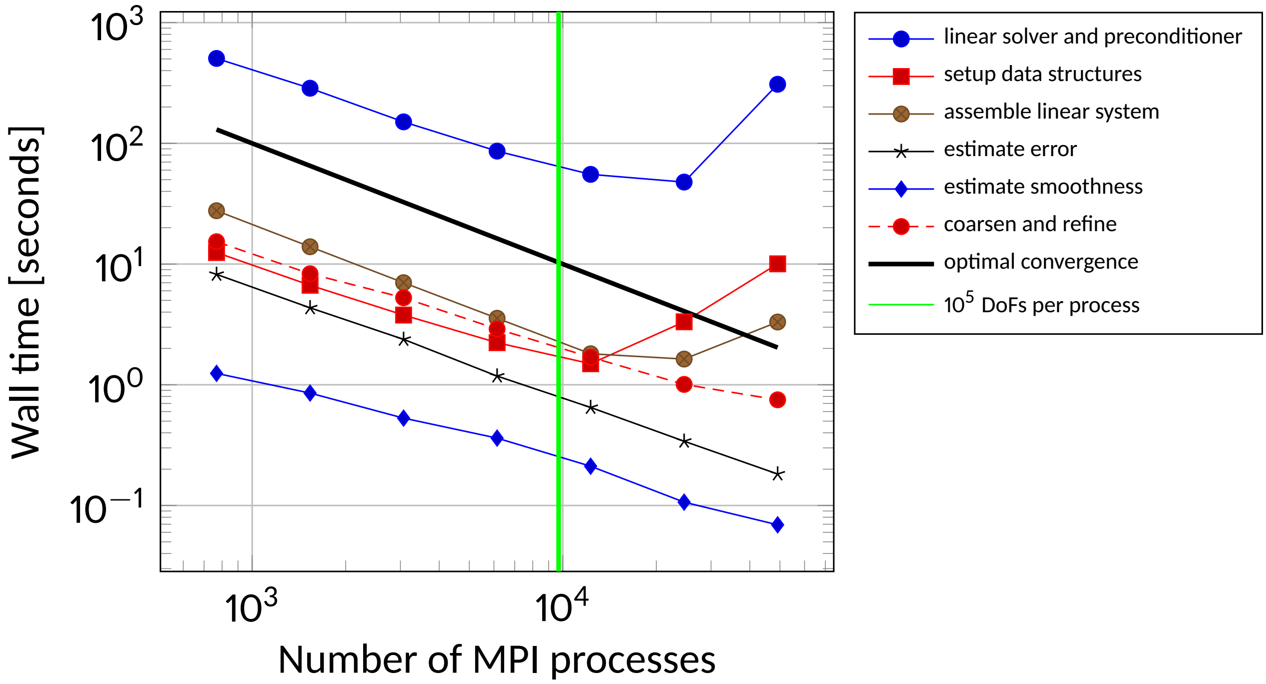


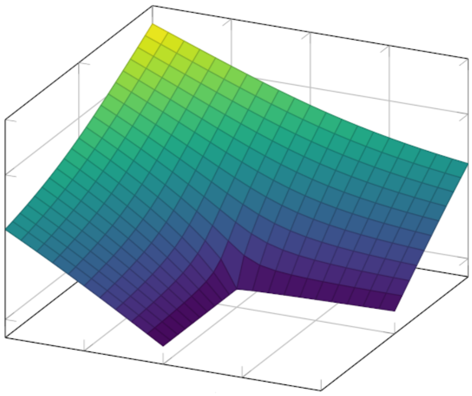
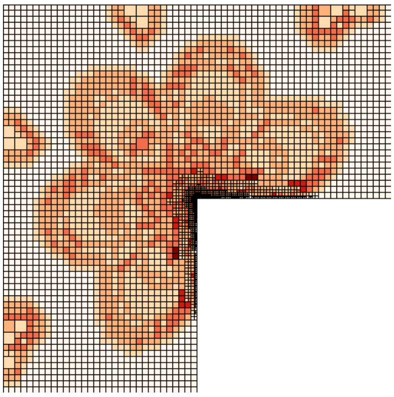
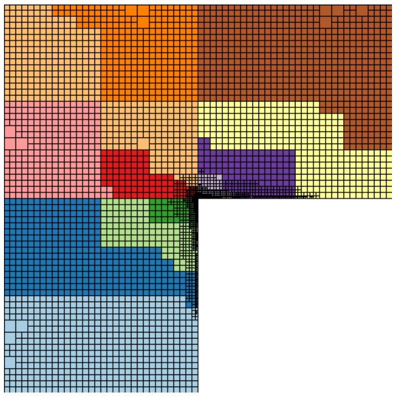
Figure: Strong scaling for fixed problem size of $\sim$ 970 million degrees of freedom

Table of contents

Summary & Outlook

- New algorithm for massively parallel hp-adaptive methods, generally applicable for any FEM software
- Reference implementation in `deal.II` involves:
  - Enumeration of degrees of freedom, independent of number of subdomains
  - Consignment of contiguous memory chunks for data transfer
  - Weighted repartitioning for load balancing
  - Selection of adaptation strategies for *hp*-FEM

- Future steps:
  - p-Multigrid methods
  - MatrixFree methods
  - Provide tutorial in `deal.II` as a manual for a broader audience
  - More applications

MASSIVELY PARALLEL HP-ADAPTIVE FEM

Bibliography

June 05, 2020 | Marc Fehling | m.fehling@fz-juelich.de |

BERGISCHE
UNIVERSITÄT
WUPPERTAL

JÜLICH
Forschungszentrum

Bibliography I

Ainsworth, M. and Senior, B. (1998).
An adaptive refinement strategy for hp-finite element computations.
Applied Numerical Mathematics, 26(1):165–178.

Arndt, D., Bangerth, W., Clevenger, T. C., Davydov, D., Fehling, M., Garcia-Sanchez, D., Harper, G., Heister, T., Heltai, L., Kronbichler, M., Kynch, R. M., Maier, M., Pelteret, J.-P., Turcksin, B., and Wells, D. (2019).
The `deal.II` Library, Version 9.1.
Journal of Numerical Mathematics, 27(4):203–213.

Babuška, I. and Guo, B. (1996).
Approximation properties of the h-p version of the finite element method.
Computer Methods in Applied Mechanics and Engineering, 133(3):319–346.

Bibliography II

Babuška, I. and Suri, M. (1990).
The p- and h-p versions of the finite element method, an overview.
Computer Methods in Applied Mechanics and Engineering, 80(1):5–26.

Bangerth, W., Burstedde, C., Heister, T., and Kronbichler, M. (2012).
Algorithms and Data Structures for Massively Parallel Generic Adaptive Finite Element Codes.
ACM Transactions on Mathematical Software, 38(2):14:1–14:28.

Bangerth, W. and Kayser-Herold, O. (2009).
Data Structures and Requirements for Hp Finite Element Software.
ACM Transactions on Mathematical Software, 36(1):4:1–4:31.

Burstedde, C. (2018).
Parallel tree algorithms for AMR and non-standard data access.
arXiv e-prints, page arXiv:1803.08432.

Bibliography III

Eibner, T. and Melenk, J. M. (2007).
An adaptive strategy for *hp*-FEM based on testing for analyticity.
Computational Mechanics, 39(5):575–595.

Houston, P. and Süli, E. (2005).
A note on the design of hp-adaptive finite element methods for elliptic partial differential equations.
Computer Methods in Applied Mechanics and Engineering, 194(2):229–243.

Kelly, D. W., De S. R. Gago, J. P., Zienkiewicz, O. C., and Babuška, I. (1983).
A posteriori error analysis and adaptive processes in the finite element method: Part I–Error Analysis.
International Journal for Numerical Methods in Engineering, 19(11):1593–1619.

Bibliography IV

Mavriplis, C. (1994).
Adaptive mesh strategies for the spectral element method.
Computer Methods in Applied Mechanics and Engineering, 116(1):77–86.

Melenk, J. M. and Wohlmuth, B. I. (2001).
On residual-based a posteriori error estimation in hp-FEM.
Advances in Computational Mathematics, 15(1):311–331.

Mitchell, W. F. and McClain, M. A. (2014).
A Comparison of hp-Adaptive Strategies for Elliptic Partial Differential Equations.
ACM Transactions on Mathematical Software, 41(1):2:1–2:39.