VSB TECHNICAL UNIVERSITY OF OSTRAVA | IT4INNOVATIONS NATIONAL SUPERCOMPUTING CENTER

Urychlení návrhu a optimalizace teplotních senzorů s využitím HPC technologií

Tomáš Kozubek

IT4Innovatins, VŠB-TUO

Konference e-infrastruktury CESNET, 29-30. ledna 2019

https://www.it4i.cz/infoservis/publikace

Contract research 2018

ŠKODA

evektor

BORCAD®

K2 atmitec

Sygic
bringing life to maps

esa

SIGMA®

DHI

varroc EXCELLENCE

mySASY

ITA technology & software

IXPERTA

brose

itica

Continental

T · · Mobile ·

BAYNCORE

DLR

ÚHÚL BRANDÝS N.LABEM

INVENT MEDICAL

DOOSAN
Doosan Škoda Power

insel

SIEMENS
New R&D centre in Ostrava

Motivation for Exascale - Digital Twin Technology

Complex nonlinear multiphysical and multiscale problem – electric motor

- Electric fields
- Electromagnetism
- Heat transfer
    - heat generated by magnetism
    - cooling system
- Structural Mechanics
    - structural integrity
    - vibration from motion
    - high speed motors
    - influenced by electromagnetism
- Active cooling system
    - fluid flow
- Acoustic
    - generated by fluid flow
    - generated by electromagnetism
    - generated by vibrations

Multiphysics simulations

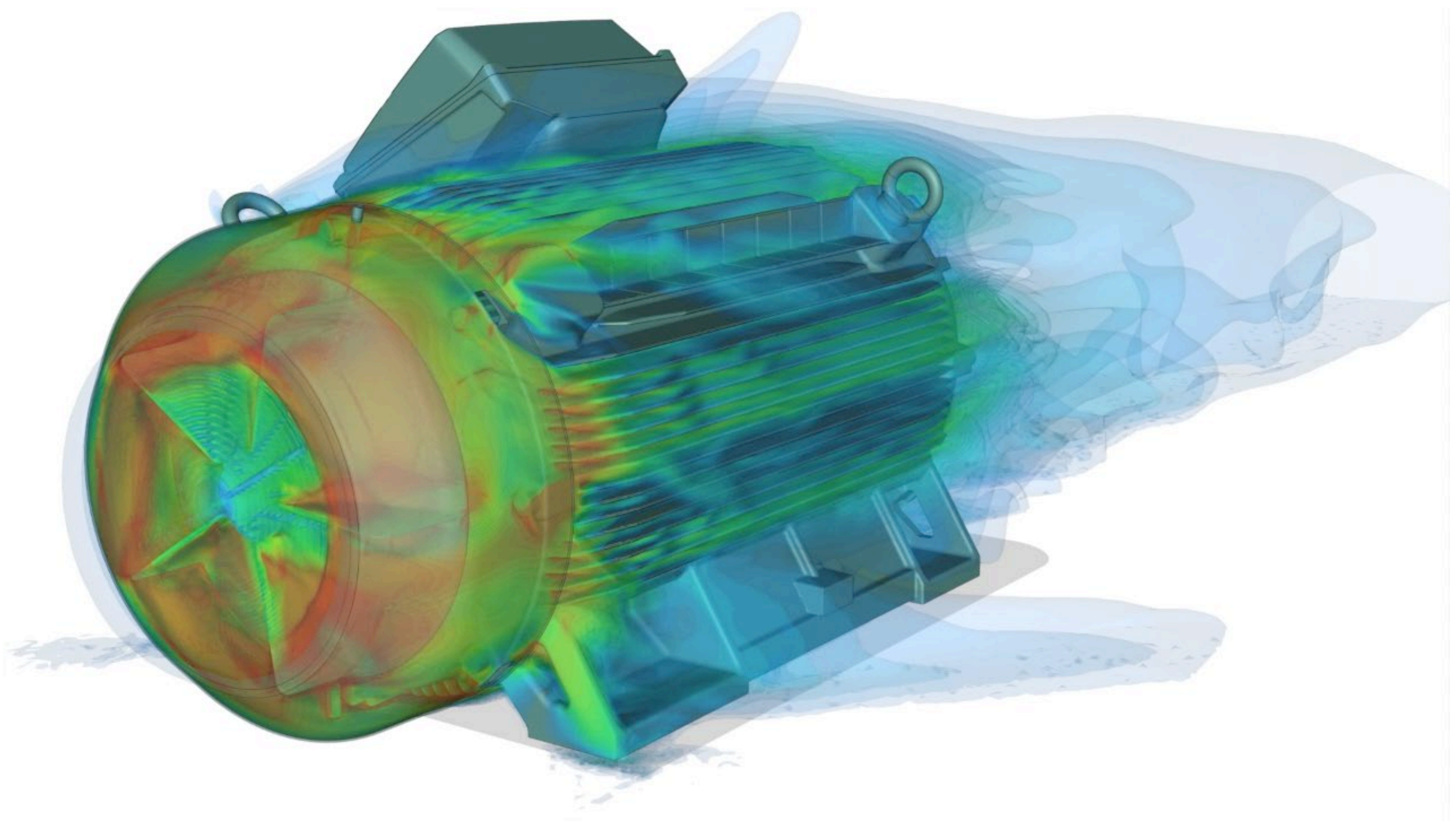
Simulation results
- Complex velocity and pressure fields
    - Ventilation looses
    - Cooling efficiency

One simulation
- 35 hours / 1200 Cores on Salomon
- “Standard” powerful workstation
- 256GB RAM 30 Cores 5 ~2 months!

What we need for optimization?
- Geometry morphing
- different number of ribs
- different fan shape
- boundary conditions variation
- …

Lead’s to 1000’s simulations
One cluster isn't enough

Full transient CFD simulation of the active cooling system



Software for engineering simulations

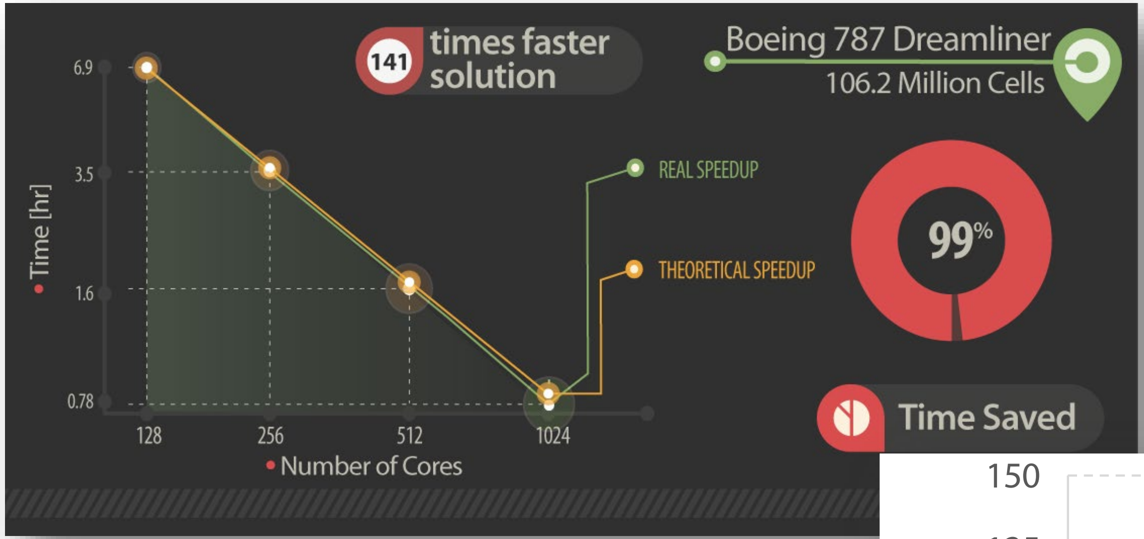
- Creating complex model, computational mesh, boundary condition definition, material models – pre-processing
- Solution by numerical methods
- Results analysis – post-processing

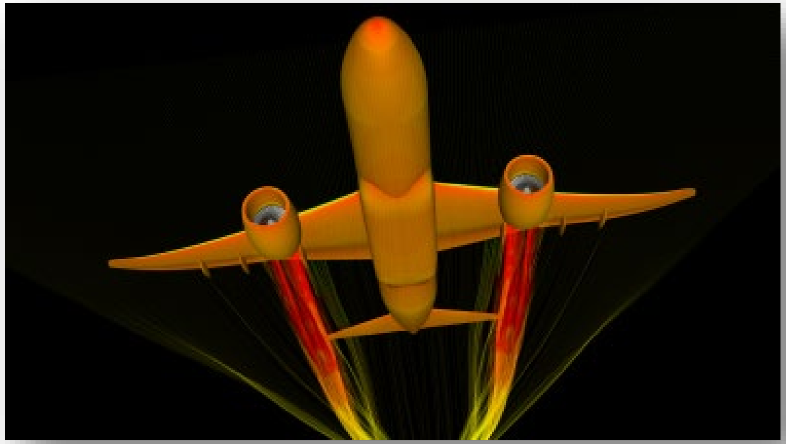
1. Open source: OpenFOAM, Code Saturne, SU^2, Elmer, Dune, …
https://www.cfd-online.com/Wiki/Codes#Free_codes

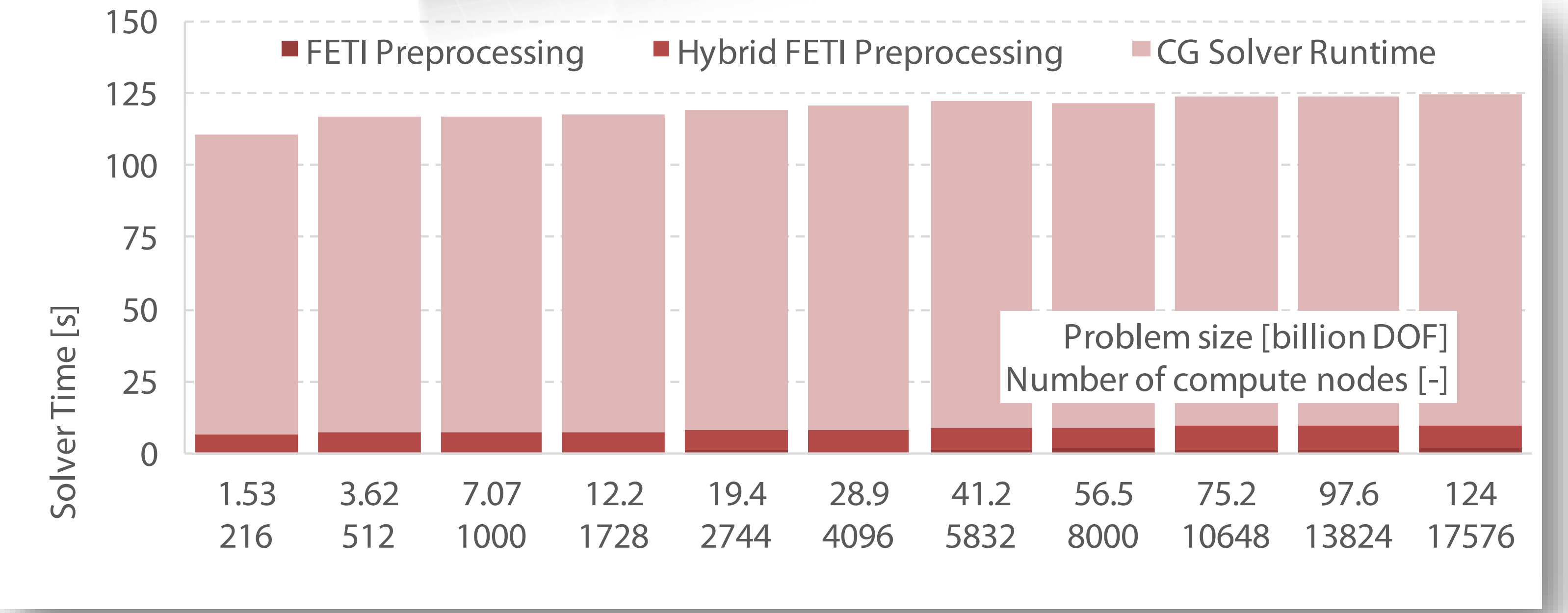
2. Commercial codes: ANSYS CFD, FLUENT, CFX, StarCCM+, COMSOL, …
https://www.cfd-online.com/Wiki/Codes#Commercial_codes

3. Or try to write it in your own way
Matlab, R, Octave, Python, C, C++, Fortran, …

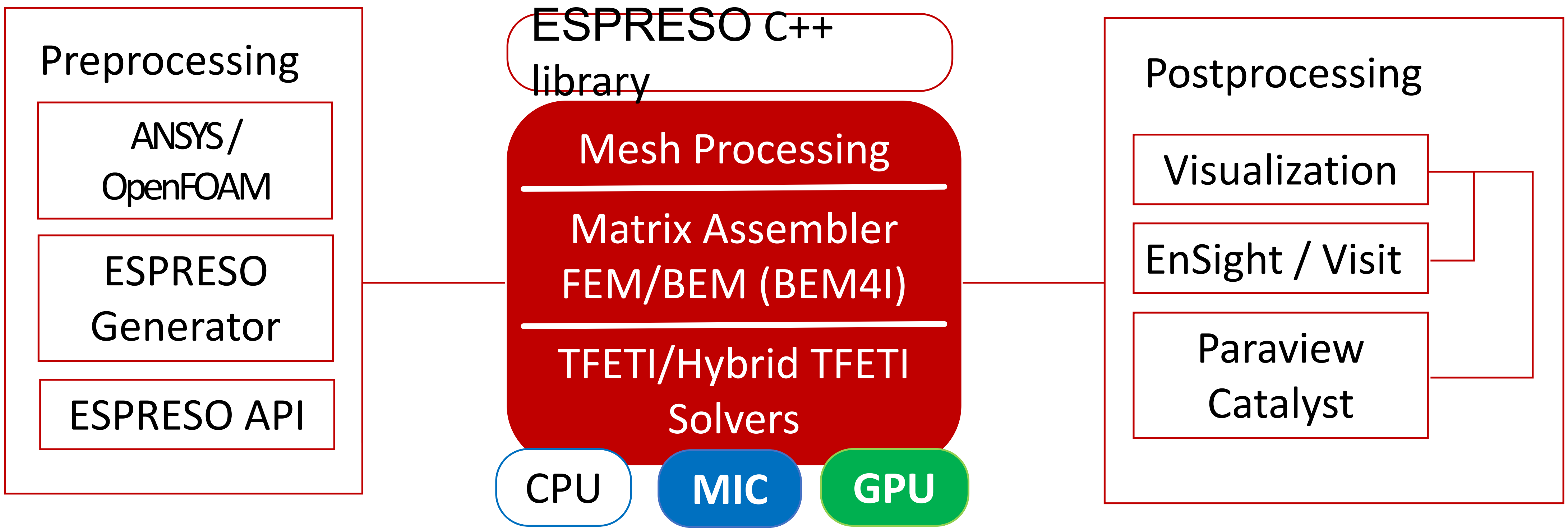
Scalable algorithms development







ESPRESO parallel solver

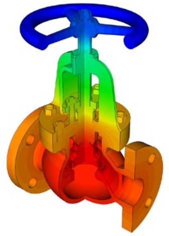


MPI, cilk++(or openMP), MKL, METIS, PARDISO(in MKL) or PARDISO SC for MIC/GPU version
Optional: SCOTCH, MUMPS, CuSolver, VTK, HYPRE interface

ESPRESO FEM Highly parallel FEM package for engineering simulations

Heat Transfer Module Capability List:

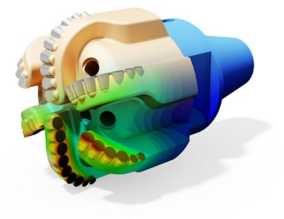
Load steps definition for combination of multiple steady-state and time dependent analyses



Transient solvers
- Generalized trapezoidal rule
- Automatic time stepping based on response frequency approach

Nonlinear solvers
- Newton Raphson – full and symmetric
- Newton Raphson with constant tangent matrices
- Line search damping
- Sub-steps definition
- Adaptive precision control for iterative solvers

Linear and quadratic finite element discretization

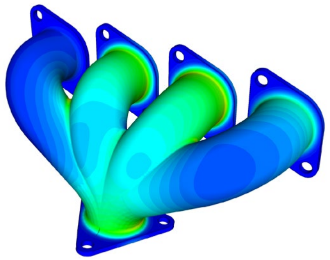


Gluing nonmatching grids by mortar discretization techniques

Full-fledged material models
- nonlinear materials
- isotropic, orthotropic and anisotropic material models
- materials for phase change

Element coordinate system definition – cartesian, polar and spherical

Temperature and time dependent boundary conditions
- linear convection
- nonlinear convection
- heat flow
- heat flux
- diffuse radiation
- heat source
- translation motion



Consistent SUPG and CAU stabilization for Translation Motion (advection), Inconsistent stabilization

Phase Change based on apparent heat capacity method

Boundary element discretization for selected physical applications

Highly parallel multilevel FETI domain decomposition based solver for billions of unknowns for symmetric and non-symmetric systems with accelerators support and combination of MPI and OpenMP techniques

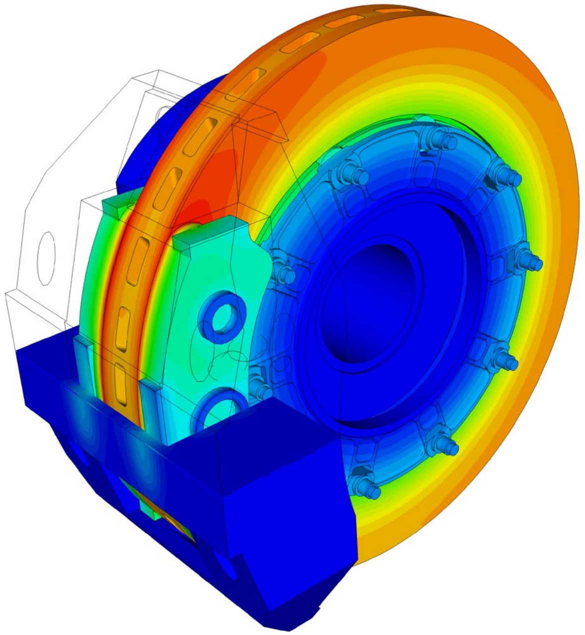
Asynchronous parallel I/O

Input mesh format from popular open source and commercial packages like OpenFOAM, ELMER or ANSYS

Output to commonly used post-processing formats, VTK and EnSight

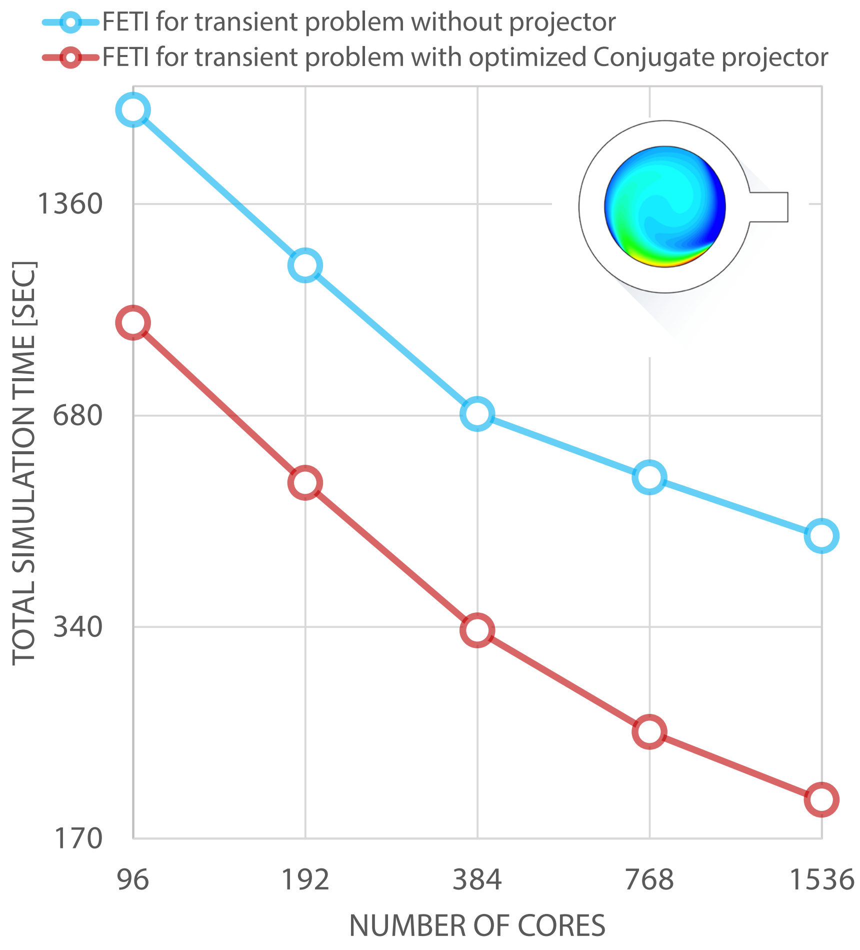
Monitoring results on selected regions for statistic and optimization toolchain

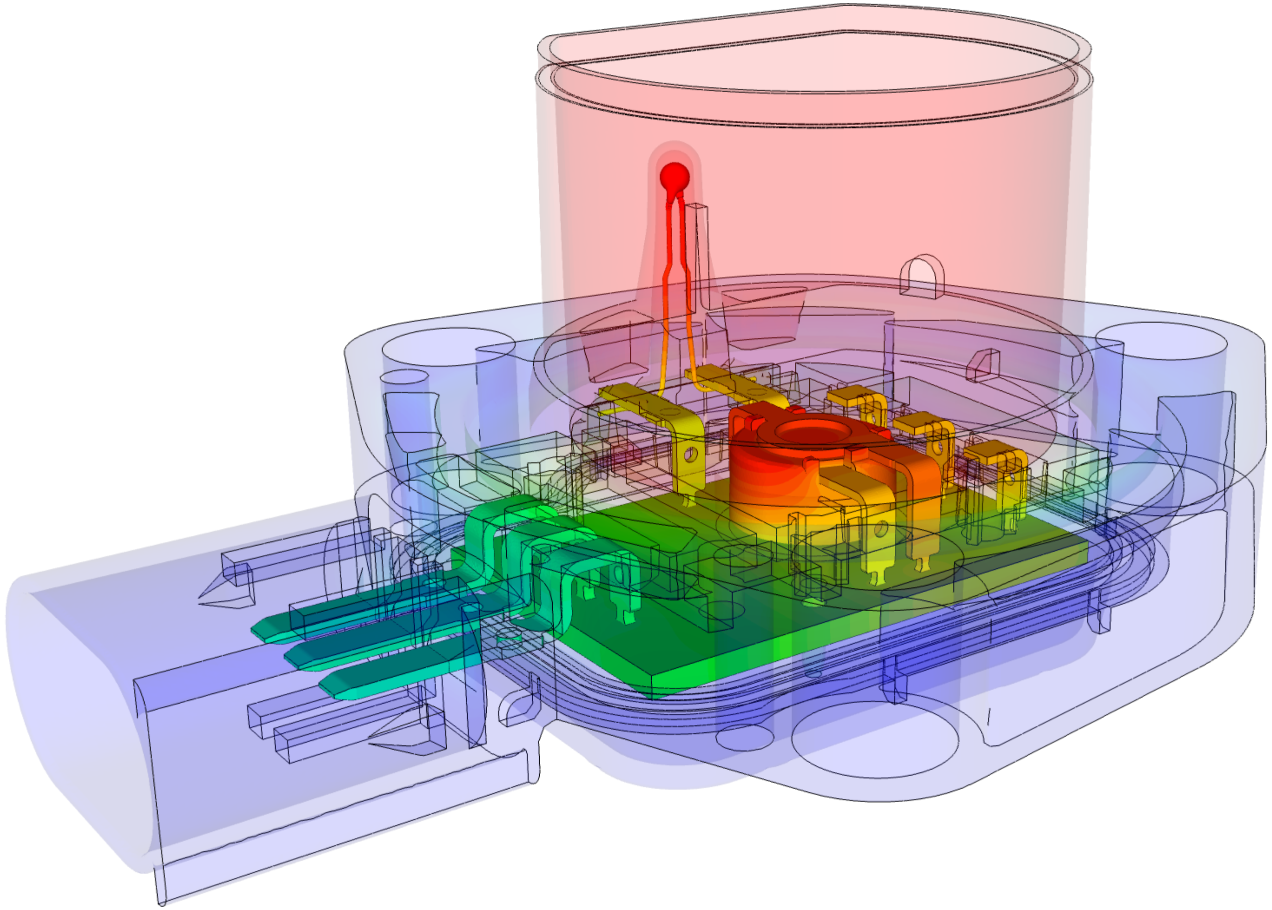
Simple text Espreso Configuration File (ecf) for setting all ESPRESO FEM solver parameters without GUI. Control each parameter in ecf file from command line



Heat transfer

Response time optimization of the USL sensor - nonlinear transient simulation

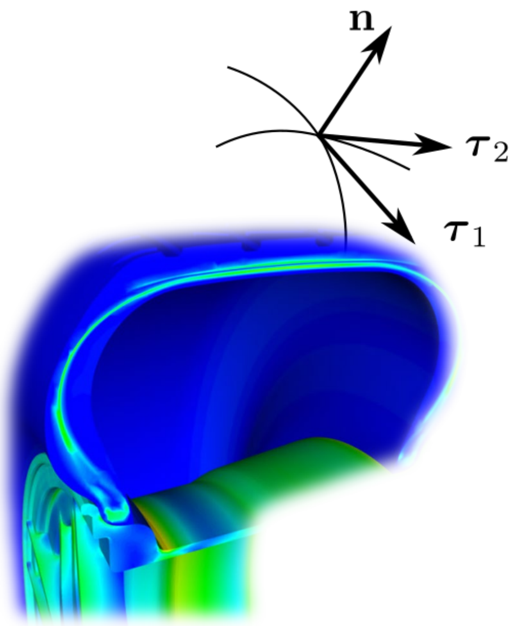


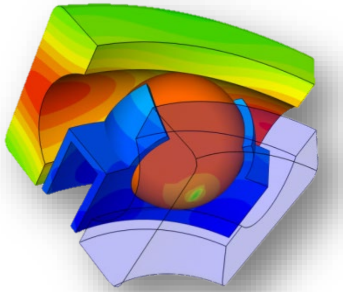


Structural mechanics

Elasticity BVP

$$
\begin{aligned}
-\mathrm{div}\boldsymbol{\sigma}^{(k)} &= \mathbf{f}^{(k)} && \text{in } \Omega^{(k)}\\
\sigma_{ij}^{(k)} &= c_{ijkl}\varepsilon_{kl}^{(k)} && \text{in } \Omega^{(k)}\\
\boldsymbol{\varepsilon}^{(k)} &= \frac{1}{2}\left(\nabla\mathbf{u} + \nabla^{\top}\mathbf{u}\right) && \text{in } \Omega^{(k)}\\
\mathbf{u}^{(k)} &= \mathbf{0} && \text{on } \Gamma_D^{(k)}\\
\boldsymbol{\sigma}^{(k)}\mathbf{n}^{(k)} &= \mathbf{t}^{(k)} && \text{on } \Gamma_N^{(k)}
\end{aligned}
$$





Terms on contact boundary

$$
\begin{aligned}
-g(\mathbf{X}) &= u_n(\mathbf{X}) - d(\mathbf{X})\\
\mathbf{t}_C^m\,\mathrm{d}\gamma_C^m &= -\mathbf{t}_C^s\,\mathrm{d}\gamma_C^s\\
\mathbf{t}_C^s &= t_n\mathbf{n} + t_{1T}\boldsymbol{\tau}_1 + t_{2T}\boldsymbol{\tau}_2
\end{aligned}
$$

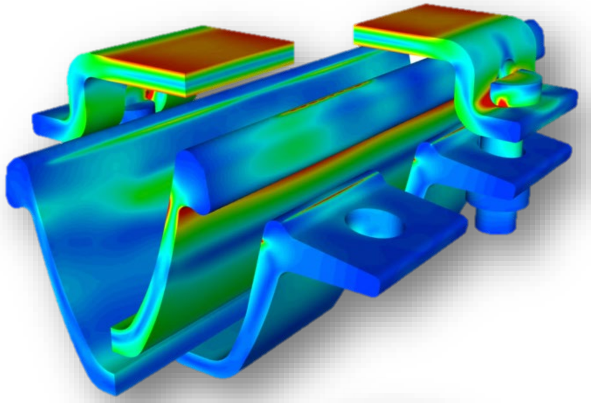
Unilateral contact (non-penetration)

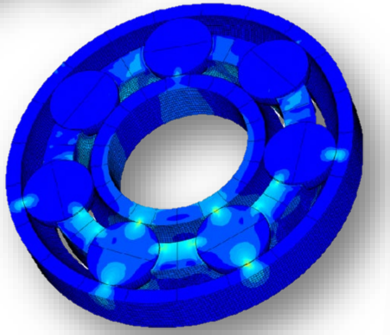
$$u_n - d \leq 0,\ t_n \leq 0,\ t_n(u_n - d) = 0$$

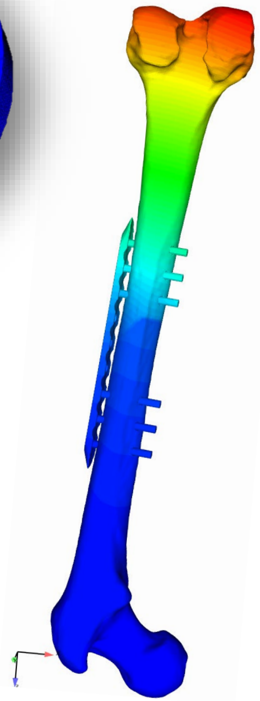
Friction (Tresca or Coulomb)

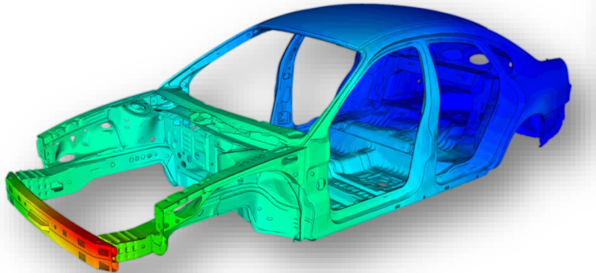
$$\|\mathbf{t}_T\|_2 \leq F,\ \begin{cases} \|\mathbf{t}_T\|_2 < F & \Rightarrow\ \mathbf{u}_T = \mathbf{0}\\ \|\mathbf{t}_T\|_2 = F & \Rightarrow\ \mathbf{u}_T = -c\mathbf{t}_T,\ c \geq 0 \end{cases}$$

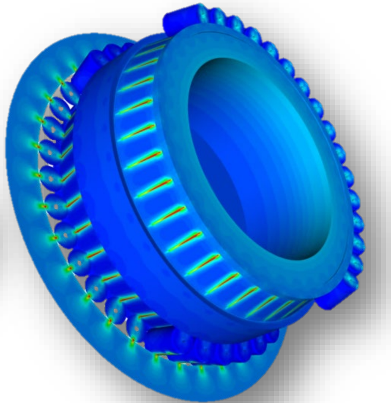
$$F = \begin{cases} sb & \ldots\ \text{Tresca}\\ \mathscr{F}|t_n| & \ldots\ \text{Coulomb} \end{cases}$$







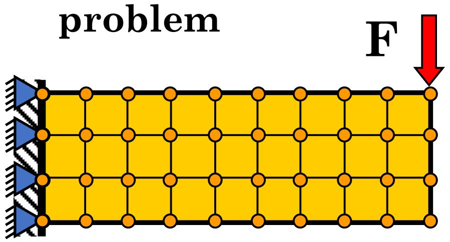


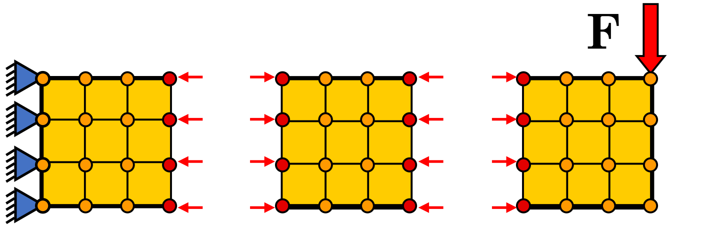
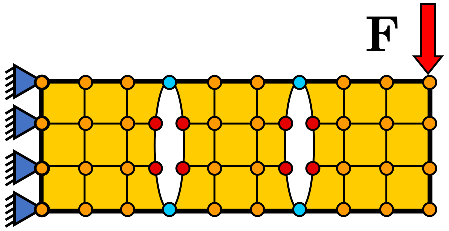
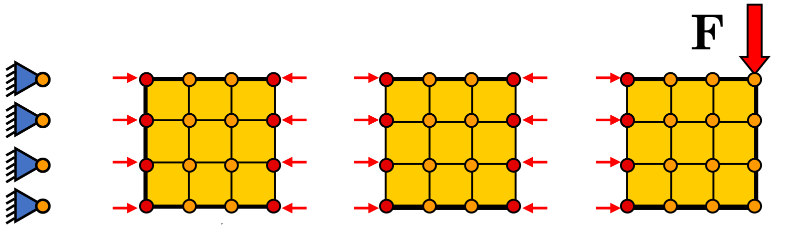


1

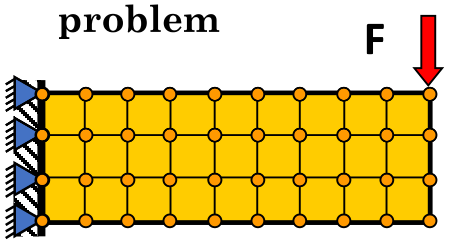
Parallelization by DDM/FETI

FETI approaches

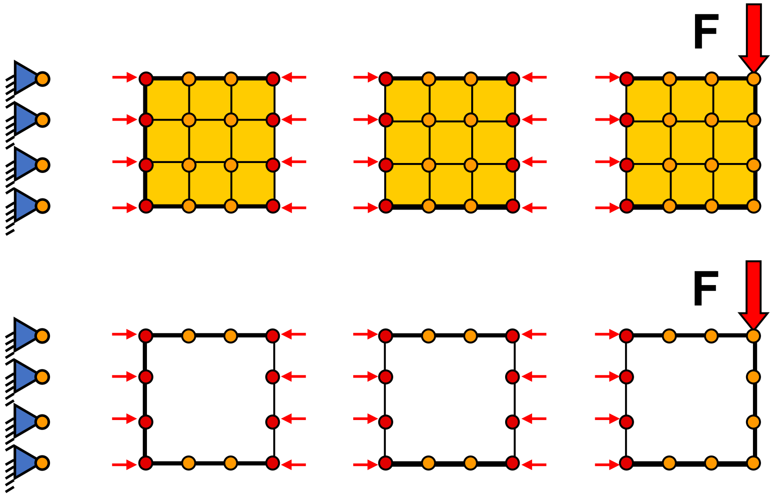


1. **FETI**		**subdomains are fixed or free but with different defects**
2. **FETI-DP**		**FETI-DP (partial splitting, nonsingular)**
3. **TFETI**		**all subdomains are free with the same defect**

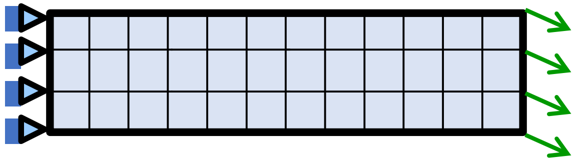
BETI approach



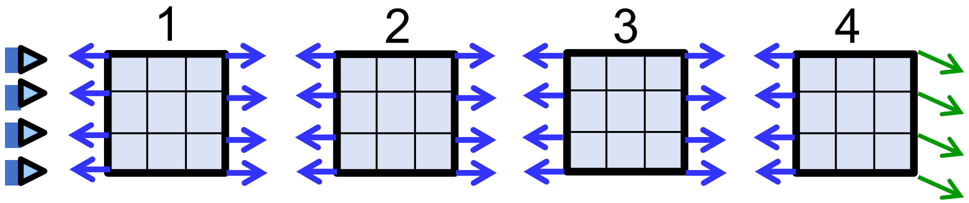
TFETI/TBETI



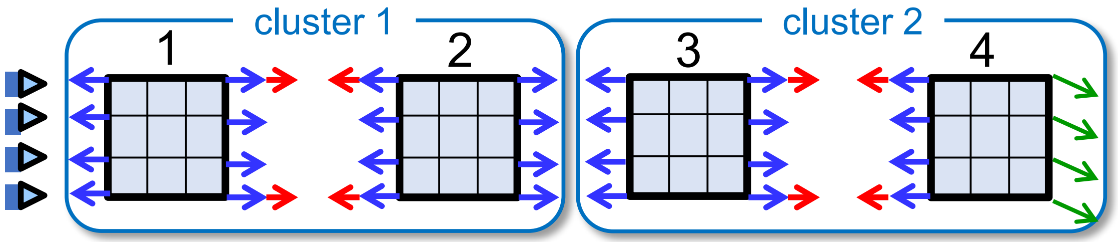
Hybrid Total FETI Method



FEM discretization

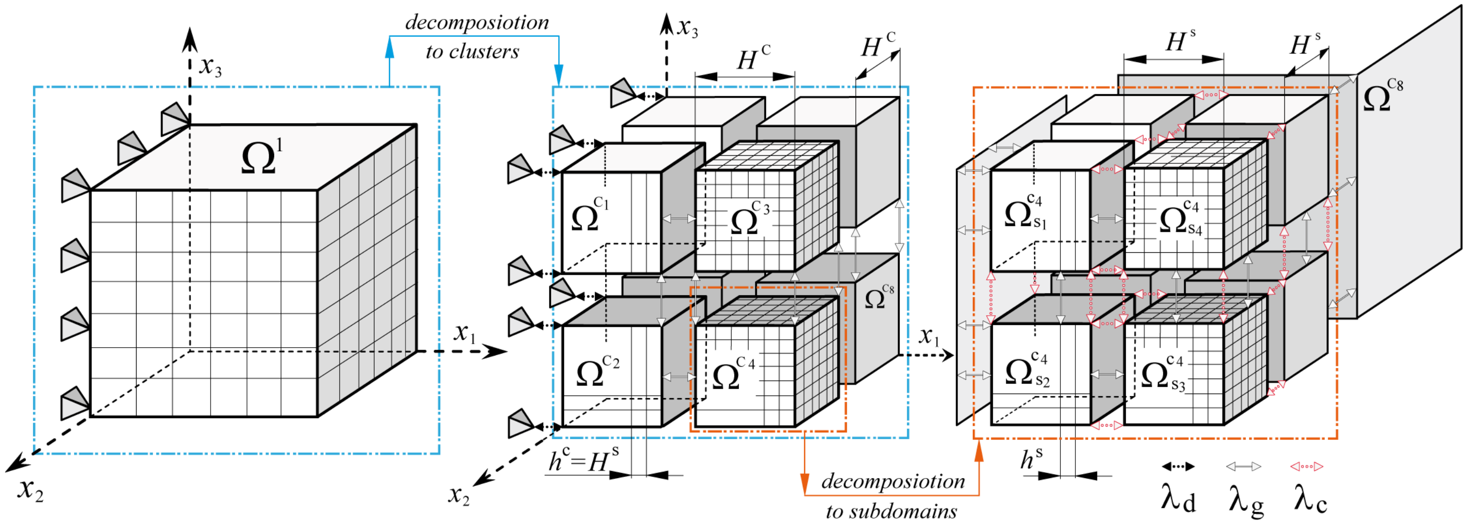


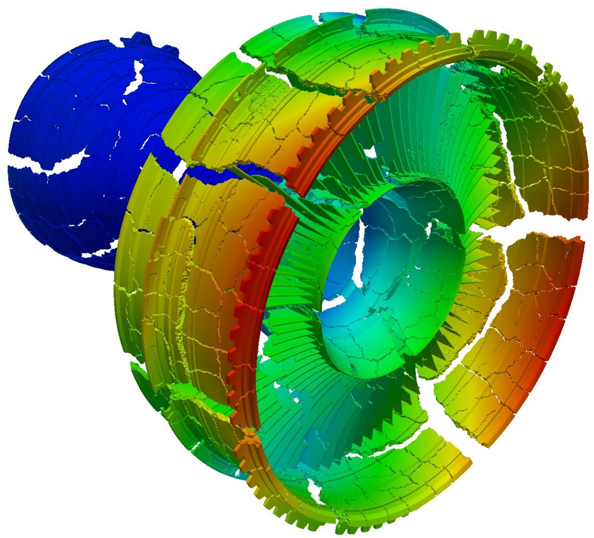
FETI method



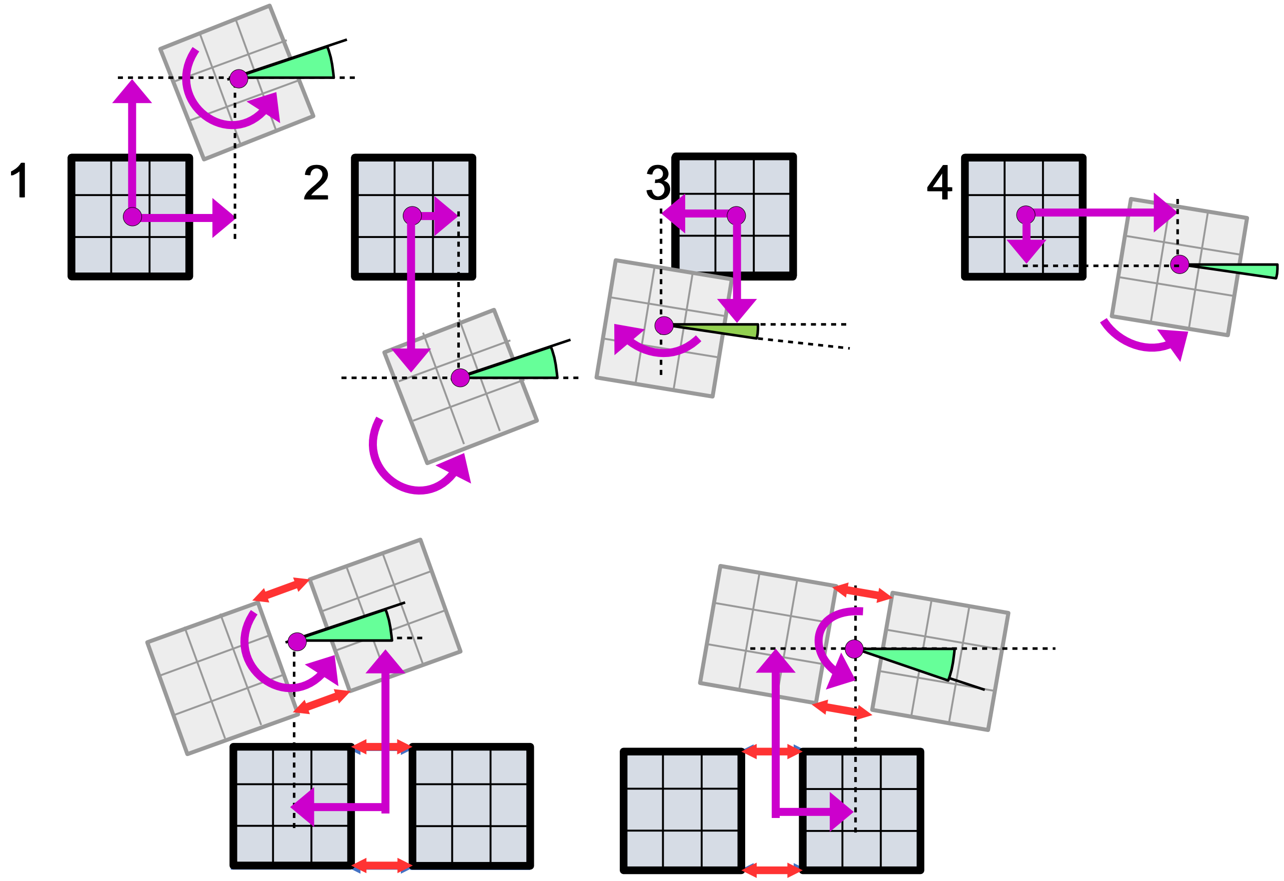
Hybrid FETI method

Hybrid Total FETI Method





Hybrid Total FETI Method



In Hybrid Total FETI method the dimension of GG^T is smaller (dim=6) compared to Total FETI (dim=12) due to additional constraints which remove local rigid body modes.

Hybrid Total FETI Method

Implementation

$$\mathbf{B}_c = \begin{pmatrix} \mathbf{B}_{c,1} & \mathbf{B}_{c,2} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{B}_{c,3} & \mathbf{B}_{c,4} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} \mathbf{B}_1 & \mathbf{B}_2 & \mathbf{B}_3 & \mathbf{B}_4 \end{pmatrix}$$

additional constraints:
duplication of 'corners' Lagrange multiplicators

$$\left(\begin{array}{cccc|cc|c} \mathbf{K}_1 & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{B}_{c,1}^\top & \mathbf{O} & \mathbf{B}_1^\top \\ \mathbf{O} & \mathbf{K}_2 & \mathbf{O} & \mathbf{O} & \mathbf{B}_{c,2}^\top & \mathbf{O} & \mathbf{B}_2^\top \\ \mathbf{O} & \mathbf{O} & \mathbf{K}_3 & \mathbf{O} & \mathbf{O} & \mathbf{B}_{c,3}^\top & \mathbf{B}_3^\top \\ \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{K}_4 & \mathbf{O} & \mathbf{B}_{c,4}^\top & \mathbf{B}_4^\top \\ \hline \mathbf{B}_{c,1} & \mathbf{B}_{c,2} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{B}_{c,3} & \mathbf{B}_{c,4} & \mathbf{O} & \mathbf{O} & \mathbf{O} \\ \hline \mathbf{B}_1 & \mathbf{B}_2 & \mathbf{B}_3 & \mathbf{B}_4 & \mathbf{O} & \mathbf{O} & \mathbf{O} \end{array}\right) \left(\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \\ \mathbf{u}_4 \\ \hline \boldsymbol{\lambda}_{c,1} \\ \boldsymbol{\lambda}_{c,2} \\ \hline \boldsymbol{\lambda} \end{array}\right) = \left(\begin{array}{c} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \\ \mathbf{f}_4 \\ \hline \mathbf{o} \\ \mathbf{o} \\ \hline \mathbf{c} \end{array}\right)$$

augmented KKT system
by the matrix $\mathbf{B}_c$ and λ_c

$$\left(\begin{array}{ccc|ccc|c} \mathbf{K}_1 & \mathbf{O} & \mathbf{B}_{c,1}^\top & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{B}_1^\top \\ \mathbf{O} & \mathbf{K}_2 & \mathbf{B}_{c,2}^\top & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{B}_2^\top \\ \mathbf{B}_{c,1} & \mathbf{B}_{c,2} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} \\ \hline \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{K}_3 & \mathbf{O} & \mathbf{B}_{c,3}^\top & \mathbf{B}_3^\top \\ \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{K}_4 & \mathbf{B}_{c,4}^\top & \mathbf{B}_4^\top \\ \mathbf{O} & \mathbf{O} & \mathbf{O} & \mathbf{B}_{c,3} & \mathbf{B}_{c,4} & \mathbf{O} & \mathbf{O} \\ \hline \mathbf{B}_1 & \mathbf{B}_2 & \mathbf{O} & \mathbf{B}_3 & \mathbf{B}_4 & \mathbf{O} & \mathbf{O} \end{array}\right) \left(\begin{array}{c} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \boldsymbol{\lambda}_{c,1} \\ \hline \mathbf{u}_3 \\ \mathbf{u}_4 \\ \boldsymbol{\lambda}_{c,2} \\ \hline \boldsymbol{\lambda} \end{array}\right) = \left(\begin{array}{c} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{o} \\ \hline \mathbf{f}_3 \\ \mathbf{f}_4 \\ \mathbf{o} \\ \hline \mathbf{c} \end{array}\right)$$

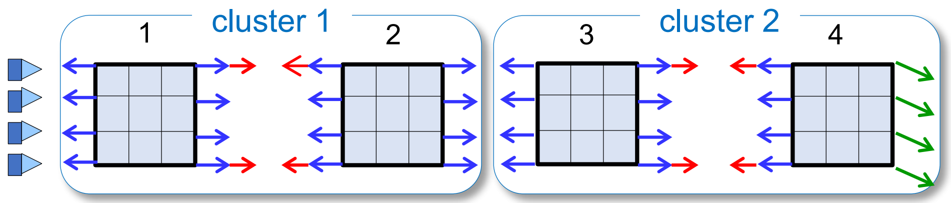
reordering according to clusters

Hybrid Total FETI Method

Implementation

$$\mathbf{B}_c = \begin{pmatrix} \mathbf{B}_{c,1} & \mathbf{B}_{c,2} & \mathbf{O} & \mathbf{O} \\ \mathbf{O} & \mathbf{O} & \mathbf{B}_{c,3} & \mathbf{B}_{c,4} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} \mathbf{B}_1 & \mathbf{B}_2 & \mathbf{B}_3 & \mathbf{B}_4 \end{pmatrix}$$

additional constraints (red arrows)



The matrix $\mathbf{B_c}$ is a copy of specific rows from the matrix $\mathbf{B}$ corresponding to components of $\boldsymbol{\lambda}$ acting on the corners between subdomains 1,2, and 3,4, respectively (red arrows).

Hybrid Total FETI Method

Modified KKT system is solvable by the same family of algorithms like FETI (Preconditioned Conjugate Projected Gradient method, …).

Size of $\tilde{G}\tilde{G}^T$ (or defect of global stiffness matrix) can be **significantly smaller** than in original FETI approach ...

$$\begin{pmatrix} \tilde{\mathbf{F}} & \tilde{\mathbf{G}} \\ \tilde{\mathbf{G}}^\top & \mathbf{O} \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{\lambda}} \\ \tilde{\boldsymbol{\alpha}} \end{pmatrix} = \begin{pmatrix} \tilde{\mathbf{d}} \\ \tilde{\mathbf{e}} \end{pmatrix}$$

$$\tilde{\mathbf{K}} = \mathrm{diag}(\tilde{\mathbf{K}}_1, \tilde{\mathbf{K}}_2)$$

$$\tilde{\mathbf{R}}^\top = (\tilde{\mathbf{R}}_1^\top, \tilde{\mathbf{R}}_2^\top)$$

$$\tilde{\mathbf{B}} = (\tilde{\mathbf{B}}_1, \tilde{\mathbf{B}}_2)$$

$$\tilde{\mathbf{F}} = \tilde{\mathbf{B}}\tilde{\mathbf{K}}^+\tilde{\mathbf{B}}^\top$$

$$\tilde{\mathbf{d}} = \tilde{\mathbf{B}}\tilde{\mathbf{K}}^+\tilde{\mathbf{f}} - \tilde{\mathbf{c}}$$

$$\tilde{\mathbf{G}} = -\tilde{\mathbf{B}}\tilde{\mathbf{R}}$$

$$\tilde{\mathbf{e}} = -\tilde{\mathbf{R}}^\top\tilde{\mathbf{f}}^\top$$

2

Pipelining

Bi-Conjugate Gradients Stabilized (BiCGStab)

Algorithm 4 Standard BiCGStab

function BICGSTAB(A, b, x_0)
  $r_0 := b - Ax_0$; $p_0 := r_0$
  for $i = 0, \ldots$ **do**
    $s_i := Ap_i$
    compute (r_0, s_i)
    $\alpha_i := (r_0, r_i) \,/\, (r_0, s_i)$
    $q_i := r_i - \alpha_i s_i$
    $y_i := Aq_i$
    compute (q_i, y_i) ; (y_i, y_i)
    $\omega_i := (q_i, y_i) \,/\, (y_i, y_i)$
    $x_{i+1} := x_i + \alpha_i p_i + \omega_i q_i$
    $r_{i+1} := q_i - \omega_i y_i$
    compute (r_0, r_{i+1})
    $\beta_i := (\alpha_i/\omega_i)\,(r_0, r_{i+1}) \,/\, (r_0, r_i)$
    $p_{i+1} := r_{i+1} + \beta_i\,(p_i - \omega_i s_i)$
  end for
end function

dot-prod
SpMV
axpy

Traditional BiCGStab:

(non-preconditioned)

Global communication

- 3 global reduction phases

Semi-local communication

- 2 non-overlapping SpMVs

Local communication

- 4 axpy(-like) operations

General two-step framework for deriving pipelined Krylov methods:

Step 1. **Avoiding communication:** *merge global reductions*

Step 2. **Hiding communication:** *overlap SpMVs & global reductions*

Pipelined BiCGStab (p-BiCGStab)

Algorithm 6 Pipelined BiCGStab

function PIPE-BICGSTAB(A, b, x_0)
  $r_0 := b - Ax_0$; $w_0 := Ar_0$; $t_0 := Aw_0$;
  for $i = 0, \dots$ **do**
    $p_i := r_i + \beta_{i-1}\,(p_{i-1} - \omega_{i-1} s_{i-1})$
    $s_i := w_i + \beta_{i-1}\,(s_{i-1} - \omega_{i-1} z_{i-1})$
    $z_i := t_i + \beta_{i-1}\,(z_{i-1} - \omega_{i-1} v_{i-1})$
    $q_i := r_i - \alpha_i s_i$
    $y_i := w_i - \alpha_i z_i$
    compute (q_i, y_i) ; (y_i, y_i)
    $\omega_i := (q_i, y_i) \,/\, (y_i, y_i)$
    overlap $v_i := Az_i$
    $x_{i+1} := x_i + \alpha_i p_i + \omega_i q_i$
    $r_{i+1} := q_i - \omega_i y_i$
    $w_{i+1} := y_i - \omega_i\,(t_i - \alpha_i v_i)$
    compute (r_0, r_{i+1}) ; (r_0, w_{i+1}) ; (r_0, s_i) ; (r_0, z_i)
    $\beta_i := (\alpha_i / \omega_i)\,(r_0, r_{i+1}) \,/\, (r_0, r_i)$
    $\alpha_{i+1} := (r_0, r_{i+1}) \,/\, ((r_0, w_{i+1}) + \beta_i\,(r_0, s_i) - \beta_i \omega_i\,(r_0, z_i))$
    overlap $t_{i+1} := Aw_{i+1}$
  end for
end function

dot-prod
SpMV
axpy

p-BiCGStab:

(non-preconditioned)

Global communication

- 2 global red. phases (vs. 3)

Semi-local communication

- 2 overlapping SpMVs

Local communication

- 8 axpy(-like) operations (vs. 4)

Status after Step 2: both global comm. phases are overlapped with SpMV computations ('hidden'), at the cost of 4 additional axpys

Pipelined BiCGStab vs. Improved BiCGStab

L.T. Yang and R.P. Brent. The improved BiCGStab method for large and sparse unsymmetric linear systems on parallel distributed memory architectures. In Proceedings of the Fifth International Conference on Algorithms and Architectures for Parallel Processing, pp. 324–328, IEEE, 2002.

	GLRED	SPMV	Flops (AXPY + DOT-PROD)	Time (GLRED + SPMV)	Memory
BiCGStab	3	2	20	3 GLRED + 2 SPMV	7
IBiCGStab	1	2	30	1 GLRED + 2 SPMV	10
p-BiCGstab	2	2^*	38	2 max(GLRED, SPMV)	11

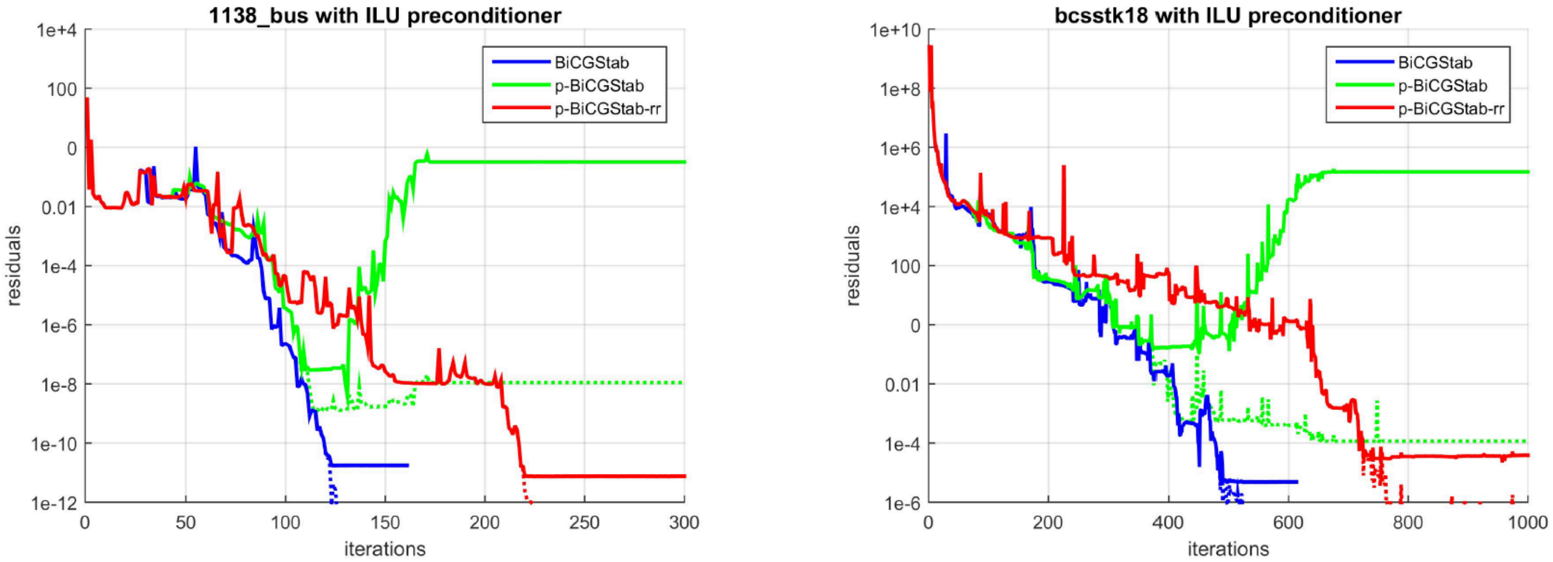
If time(GLRED) $\approx$ time(SPMV):

- speed-up factor p-BiCGStab/BiCGStab = 2.5
- speed-up factor IBiCGStab/BiCGStab = 1.66

If time(GLRED) $\gg$ time(SPMV):

- speed-up factor p-BiCGStab/BiCGStab = 2.5
- speed-up factor IBiCGStab/BiCGStab = 3.0

Robustness and attainable accuracy: p-BiCGStab-rr



Residual replacement every rr-th iteration
(non-automated, i.e. rr is a parameter of the method, but chosen large s.t. no. res. repl. is small)

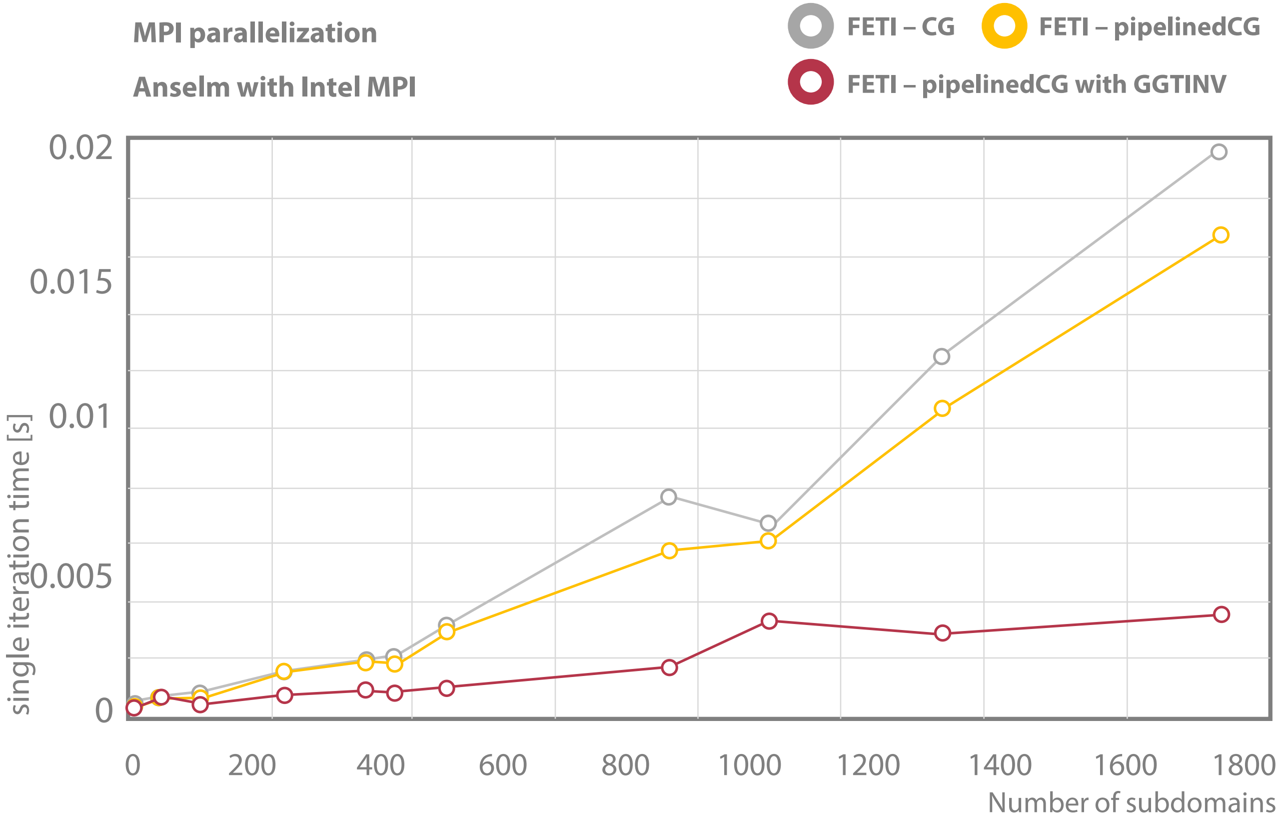
$$r_i := b - Ax_i, \qquad \hat{r}_i := M^{-1} r_i, \qquad w_i := A\hat{r}_i,$$

$$s_i := A\hat{p}_i, \qquad \hat{s}_i := M^{-1} s_i, \qquad z_i := A\hat{s}_i.$$

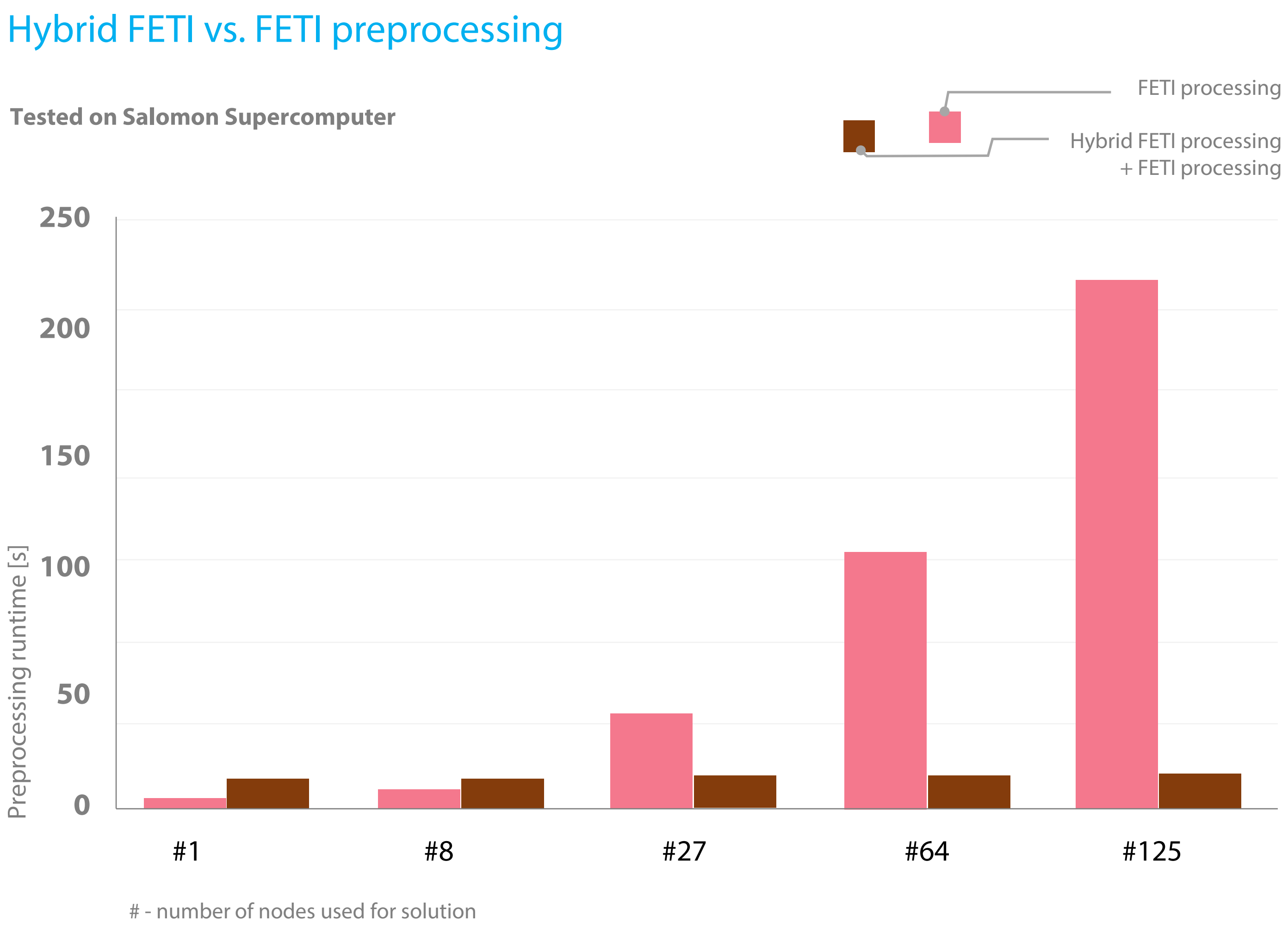
- ⊕ increased **maximal attainable accuracy**: comparable to BiCGStab level
- ⊕ increased **robustness**: negates instable true residual behaviour
- ⊖ increased **number of iterations** possible

ESPRESO FETI

Optimization of global communication for FETI Solvers



ESPRESO FETI



3

Accelerators

DOE HPC Facilities Systems

Pre-Exascale Systems				Exascale Systems
2013	2016	2018	2020	2021-2022

2013

Argonne
IBM BG/Q
Unclassified

ORNL
Cray/NVidia K20
Unclassified

LLNL
IBM BG/Q
Classified

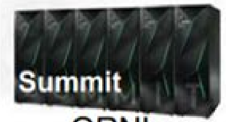
2016

Argonne
Intel/Cray KNL
Unclassified

LBNL
Cray/Intel Xeon/KNL
Unclassified

LANL/SNL
Cray/Intel Xeon/KNL
Classified

2018



ORNL
IBM/NVidia
P9/Volta
Unclassified

LLNL
IBM/NVidia
P9/Volta
Classified

2020

NERSC-9

LBNL
TBD
Unclassified

Crossroads

LANL/SNL
TBD
Classified

2021-2022

Argonne
Intel/Cray TBD
Unclassified

Frontier

ORNL
TBD
Unclassified

El Capitan

LLNL
TBD
Classified

Argonne National Laboratory
Lawrence Livermore National Laboratory
Oak Ridge National Laboratory

ESPRESO CG in FETI

Projected Conjugate Gradient in FETI

$r_0 := b - Ax_0;\ u_0 := M^{-1} r_0;\ p_0 := u_0$
for $i = 0, \ldots, m-1$ **do**
  $s := Ap_i$
  $\alpha := \langle r_i . u_i \rangle / \langle s . p_i \rangle$
  $x_{i+1} := x_i + \alpha p_i$
  $r_{i+1} := r_i - \alpha s$
  $u_{i+1} := M^{-1} r_{i+1}$
  $\beta := \langle r_{i+1} . u_{i+1} \rangle / \langle r_i . u_i \rangle$
  $p_{i+1} := u_{i+1} + \beta p_i$
end for

Pre-processing – K factorization

1.) $x = B_1^T \cdot \lambda$ - **SpMV**
2.) $y = K^{-1} \cdot x$ - **solve**
3.) $\lambda = B_1 \cdot y$ - **SpMV**
4.) stencil data exchange in λ
- MPI – Send and Recv
- OpenMP – shared mem. Vec

Pre-processing - $S_c = B_1 K^{-1} B_1^T$

1.) - **nop**
2.) $\lambda = S_c \cdot \lambda$ - **DGEMV, DSYMV**
3.) - **nop**
4.) stencil data exchange in λ
- MPI – Send and Recv
- OpenMP – shared mem. Vec

Pre-processing - $S_c = B_1 K^{-1} B_1^T \rightarrow$ GPU/MIC

1.) $\lambda \rightarrow$ GPU/MIC - **PCIe transfer from CPU**
2.) $\lambda = S_c \cdot \lambda$ - **DGEMV, DSYMV on GPU/MIC**
3.) $\lambda \leftarrow$ GPU/MIC - **PCIe transfer to CPU**
4.) stencil data exchange in λ
- MPI – Send and Recv
- OpenMP – shared mem. vec

90 – 95% of runtime spent in Ap_i

GPU acceleration of the ESPRESO Solver

0.3 - 300 million DOF Hybrid FETI CG Solver Runtime

Linear elasticity

ORNL Titan 5rd in TOP500 LIST

speedUp

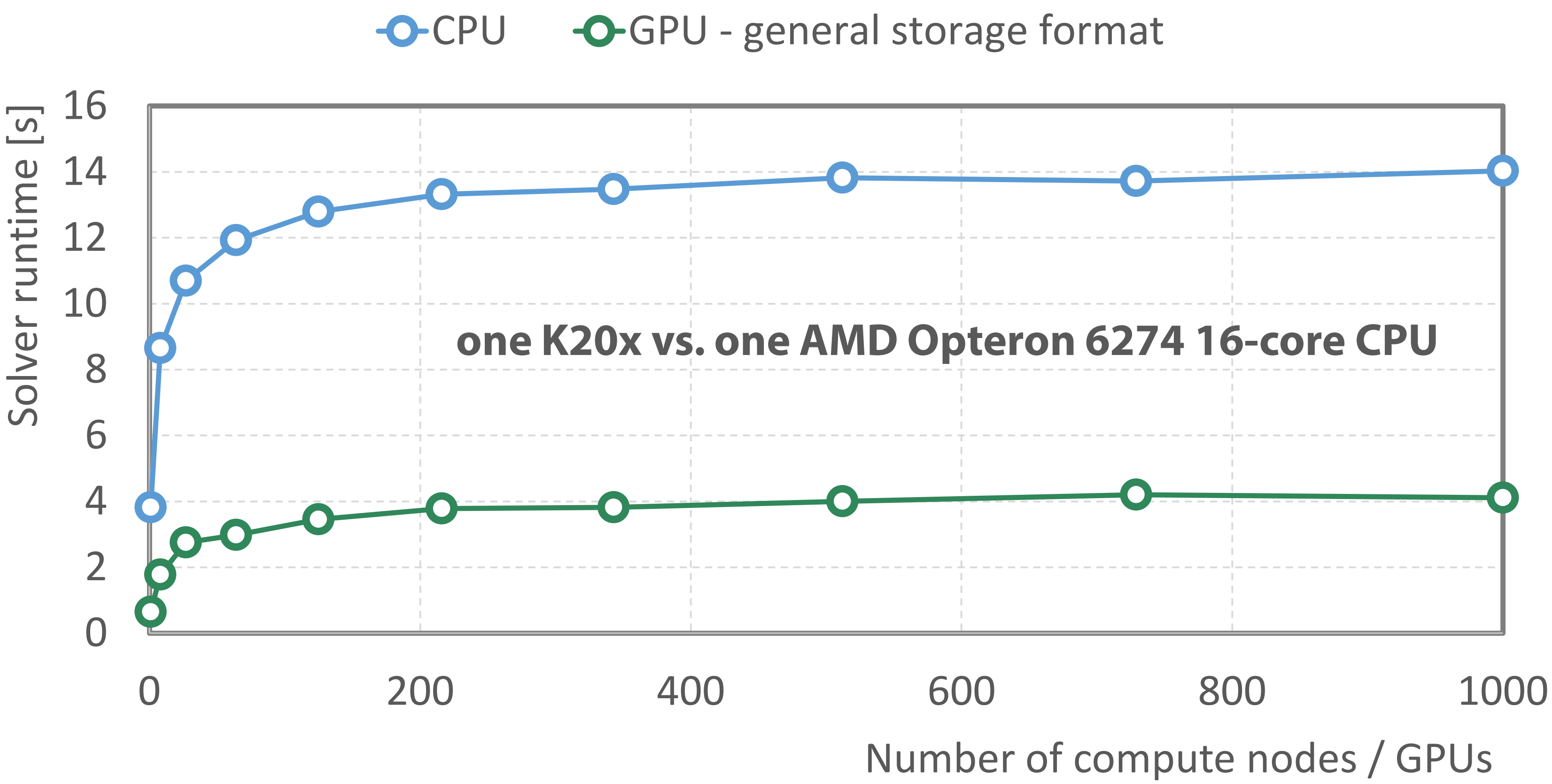
3.4

•Current work:

- support for cuDense and cuSparse solvers from CUDA toolkit
- memory optimization for symmetric local schur complements
- implementation of memory efficient methods

•Future work:

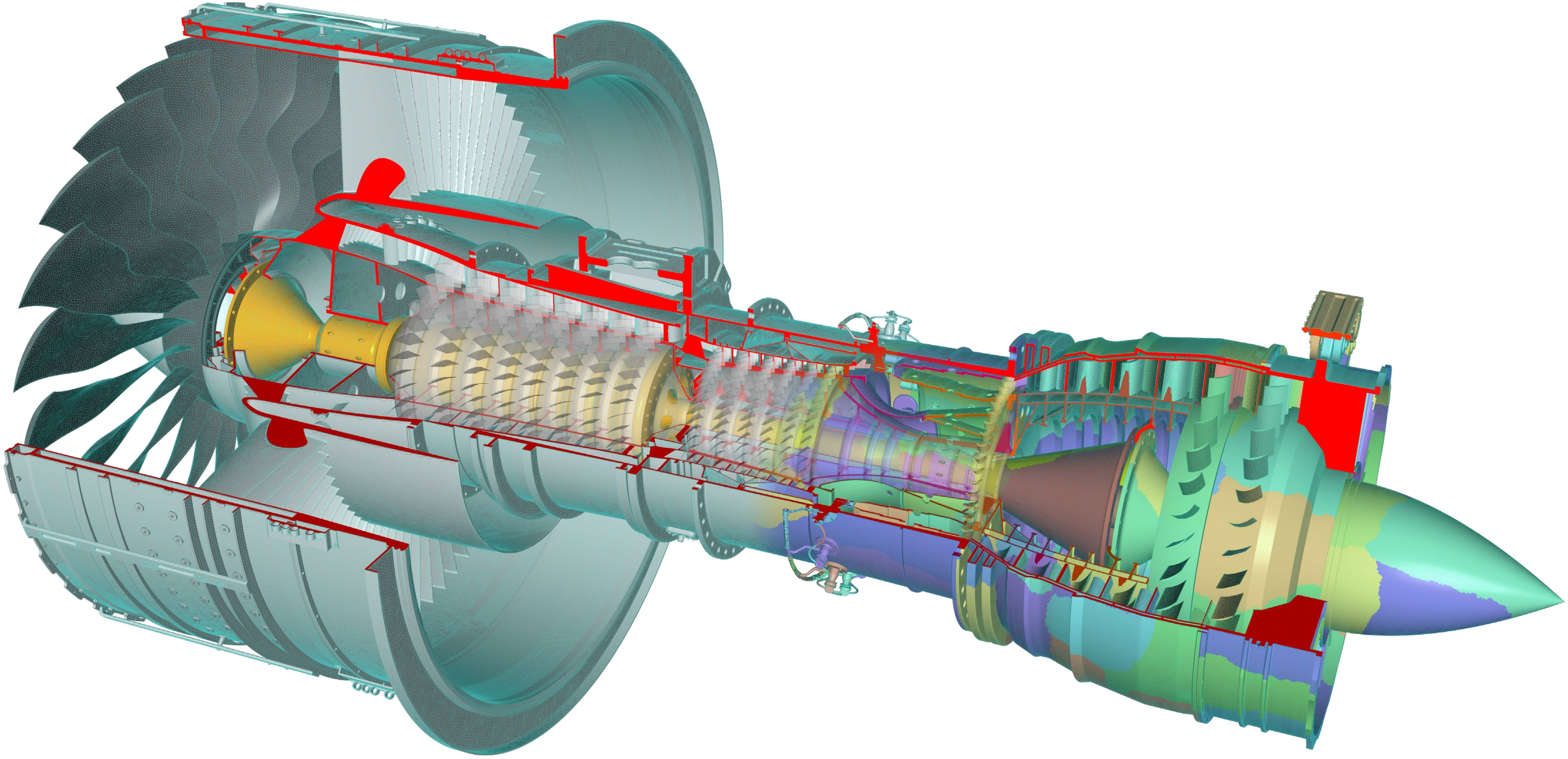
- Support for Power8/9 platforms with Pascal/Volta GPUs with NVLink



4

Parallel pre-processing

ESPRESO FEM – pre-processing

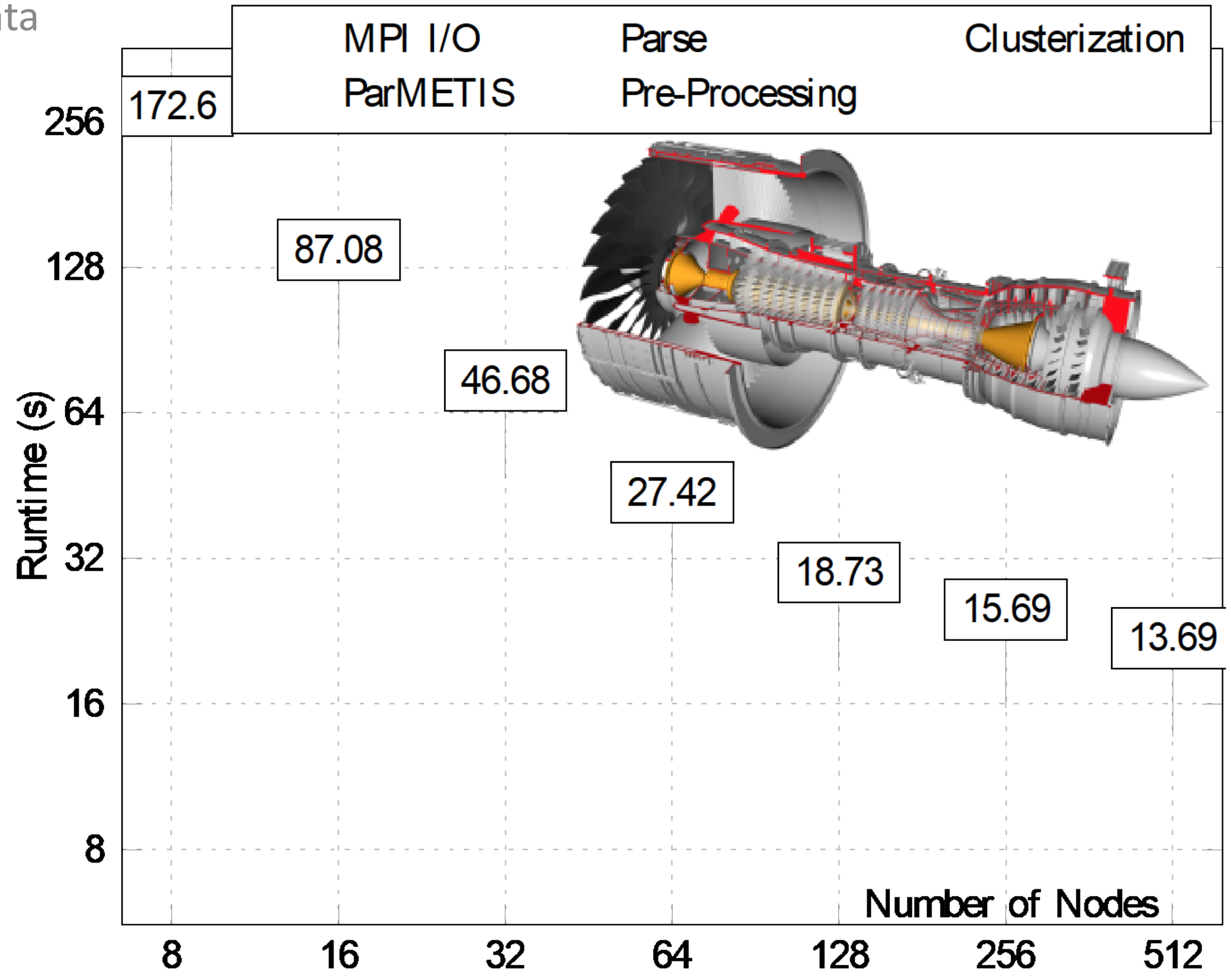


Efficient parallel loading of seq. stored unstructured mesh databases

Direct connection with commercial pre-processing tools

Highly parallel I/O of external sequentially stored data with domain decomposition techniques

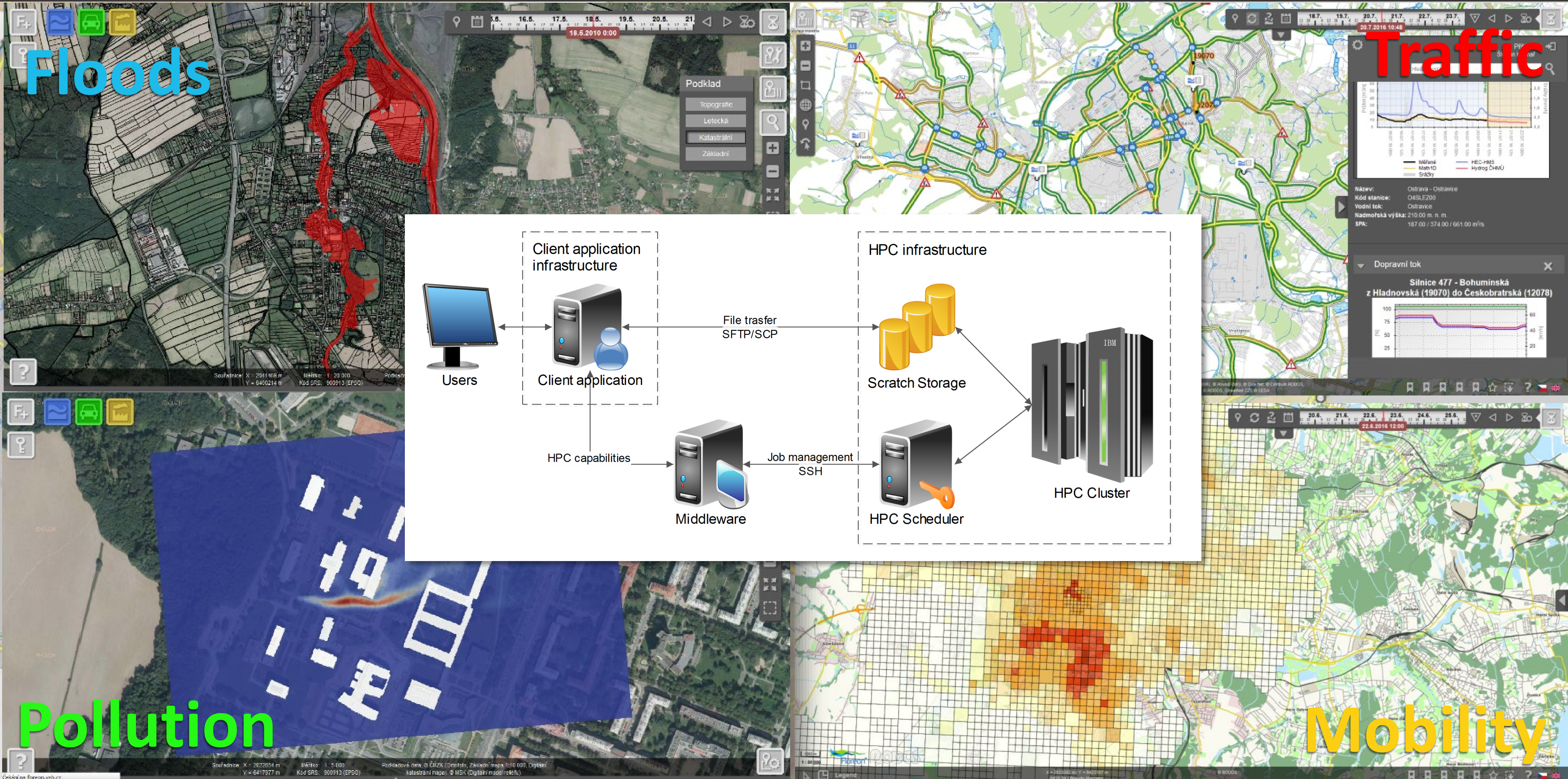
Jet Engine	
file size	142 GB
Nodes	822 M
Elements	484 M
seq. processing	7200s
par. Processing	13.69 s



5

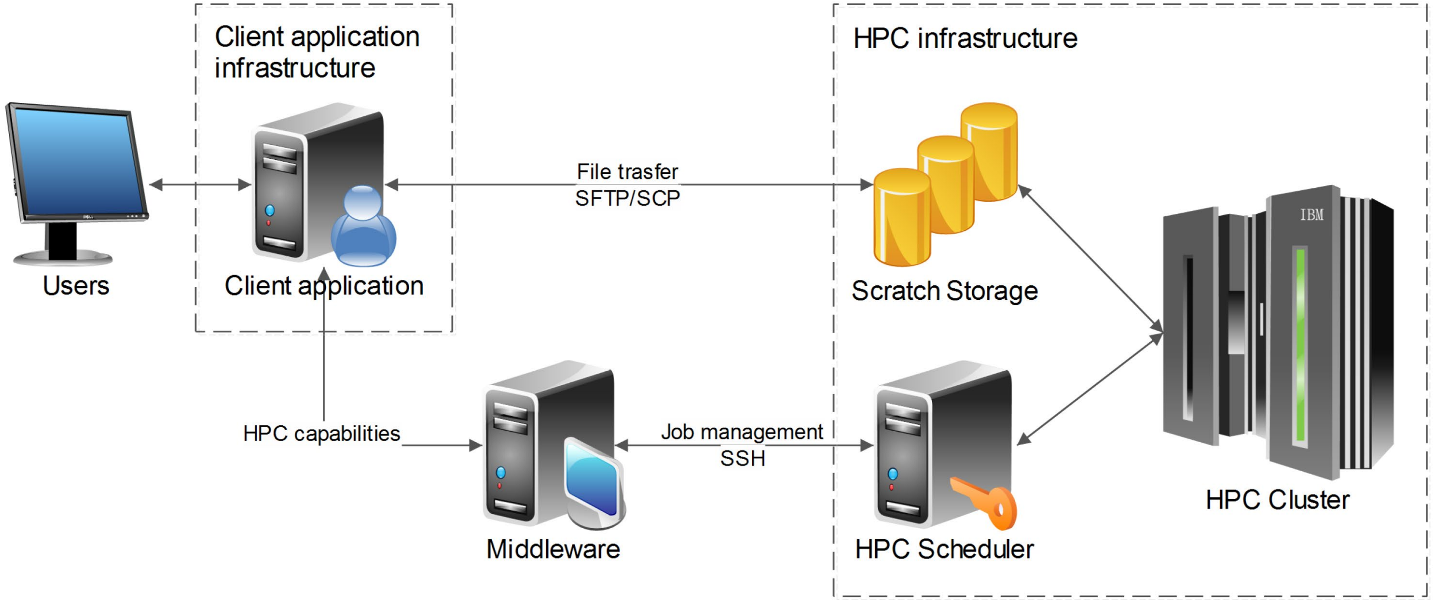
GUI and Solver as a Service

Online Solver as a Service (HPC as a Service)



HPC as a Service (HEAppE) Overview

- Providing HPC capabilities as a service to client applications and their users
- Unified interface for different operating systems and schedulers
- Authentication and authorization to provided functions
- Monitoring and reporting of executed jobs and their progress
- Current information about the state of the cluster



Thank you for attention