



PEPR NumPEX // Exa-MA

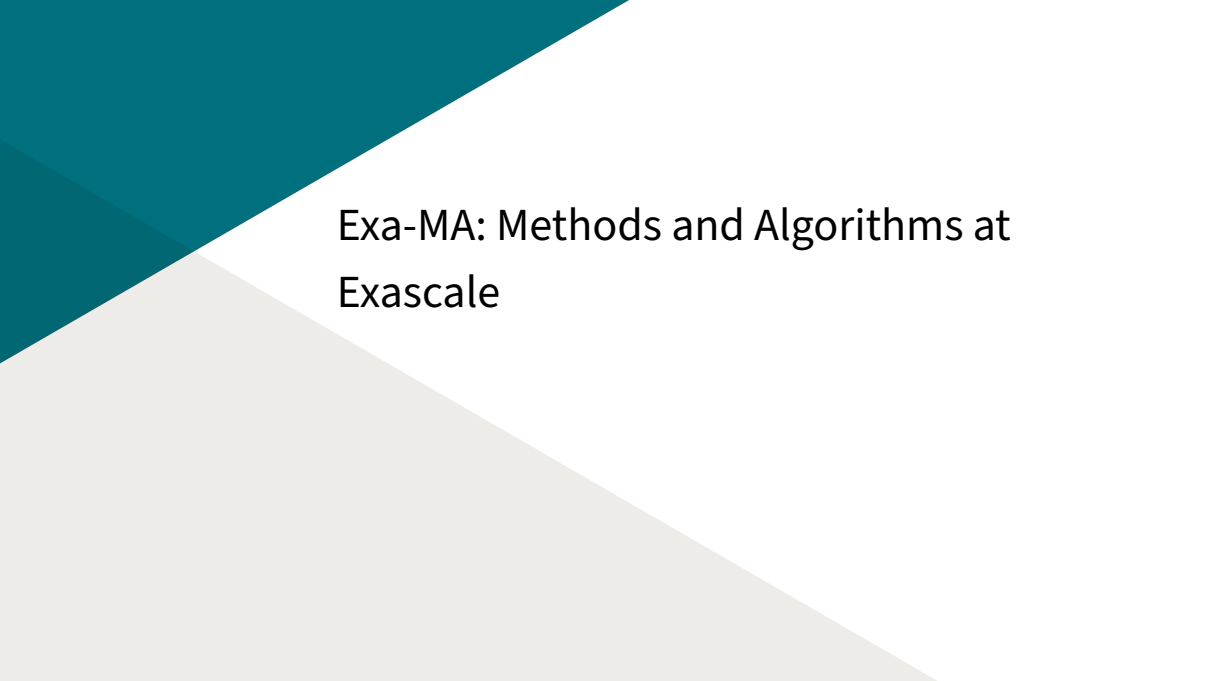
Methods and Algorithms at
ExaScale

PEPR - Genci // 5 Janvier 2023

Christophe Prud'homme & Hélène Barucq

Overview



The background of the slide is composed of two large, overlapping geometric shapes. A teal-colored shape occupies the top-left corner, while a light gray shape occupies the bottom-left corner. The rest of the slide is white.

Exa-MA: Methods and Algorithms at Exascale

Exa-MA: Methods and Algorithms at Exascale



Objectives

The main objectives are

- ▶ O1: to develop methods, algorithms, and implementations that, taking advantage of the exascale architectures, empower modeling, solving, assimilating model and data, optimizing and quantifying uncertainty, at levels that are unreachable at present;
- ▶ O2: to develop or contribute to software libraries allowing to assemble specific critical reusable components, hiding the hardware complexity and exposing only the specific methodological interface ;
- ▶ O3: to identify and co-design Methodological and Algorithmic Patterns at exascale that can be reused efficiently in large scale applications (eg in weather forecasting);
- ▶ O4: to enable AI algorithms to attain performances at exascale, exploiting the methods (O1) and the libraries (O2) developed; and
- ▶ O5: to provide demonstrators through mini-apps and proxy-apps that will be openly available and benchmarked.

Exa-MA: Methods and Algorithms at Exascale



WP1: Discretization

Objectives

- ▶ Geometric domain representations and their discrete counterparts
- ▶ Physics-based models

Specifications

- ▶ Favor high order methods to increase local compute load
- ▶ Favor non-conforming methods to reduce communication

Links

PC2-WP2/3/4, PC3-WP3

Tasks

- ▶ T1.1 Geometry and Mesh at Exascale
- ▶ T1.2 Adaptive Mesh Refinement for unstructured grids
- ▶ T1.3 Adaptive Mesh Refinement for cartesian or block grids
- ▶ T1.4 Finite Element Exascale Framework (FEEF)
- ▶ T1.5 Exploit non conforming methods for efficient parallel computing
- ▶ T1.6 Error control of time discretization for accurate space refinement
- ▶ T1.7 Efficient multimodel/multphysics coupling

Exa-MA: Methods and Algorithms at Exascale



WP2: Reduced order and AI driven methods for multi-fidelity modeling

Objectives

- ▶ Develop Reduced order methods
- ▶ Develop methods for multi-fidelity modeling

Specifications

- ▶ Leverage beyond state of the art reduce order, surrogate and machine learning methods
- ▶ Enable Multi-fidelity modeling

Tasks

- ▶ T2.1 surrogate models based on physics driven deep learning
- ▶ T2.2 PDE operator learning using NN
- ▶ T2.3 data driven model order reduction
- ▶ T2.4 non-intrusive and weakly intrusive reduced basis methods for parametrized PDEs
- ▶ T2.5 mixing low and high fidelity models
- ▶ T2.6 real-time models with super resolution methods

Links

PC2-WP2/3/4, PC3-WP3

Exa-MA: Methods and Algorithms at Exascale



WP3: Linear, Multi-linear and Coupled Solvers at Exascale

Objectives

- ▶ Focus on generic building blocks(algebraic) for informations
- ▶ Support high dimensional problems

Specifications:

- ▶ Communication avoiding algorithms
- ▶ low-precision computing
- ▶ matrix-free methods
- ▶ operator/data compression

Tasks

- ▶ T3.1 Acceleration techniques for subspace-based methods;
- ▶ T3.2 High dimensional problems ;
- ▶ T3.3 Randomization;
- ▶ T3.4 Exploiting data-sparsity and multiple precision and enabling resilience;
- ▶ T3.5 Adaptive solution strategies for exascale multiphysical and multiscale models;

Links

PC2-WP2/3/4

Exa-MA: Methods and Algorithms at Exascale



WP4: Combine data and models, inverse problems at Exascale

Objectives

- ▶ Improve existing deterministic methods
- ▶ Formulate new stochastic methods.
- ▶ Improve observation strategies.
- ▶ Implement multi-fidelity schedules

Specifications:

- ▶ combine model and data
- ▶ Enable deterministic and stochastic methods

Tasks

- ▶ T4.1 Deterministic methods
- ▶ T4.2 Stochastic methods
- ▶ T4.3 Observations
- ▶ T4.4 Taking advantage of multi-fidelity modeling
- ▶ T4.5 Challenges of multi-fidelity in inverse problems: criteria to update reduced models.

Links

PC2-WP1/2/3/4,PC3-WP3

Exa-MA: Methods and Algorithms at Exascale



WP5: Optimize at Exascale

Objective: Design and implement of large scale optimization problems

- ▶ combinatorial continuous and mixed optimization
- ▶ surrogate-based optimization
- ▶ shape optimization

Specifications / Tasks

- ▶ T5.1 Decomposition-based methods
- ▶ T5.2 Learning-based methods, e.g. surrogate models and multi-fidelity representation
- ▶ T5.3 Reduced order and ML for shape optimization
- ▶ T5.4 Auto-tuning for ML

Links

PC2-WP1/2/3/4, PC3-WP2

Exa-MA: Methods and Algorithms at Exascale



WP6: Quantify uncertainty at Exascale

Objectives

- ▶ Sensitivity analysis for dimension reduction, ranking and more generally understanding the influence of uncertain input parameters.
- ▶ Propagation of uncertainties
- ▶ Surrogate modeling for UQ
- ▶ Acceleration of the bricks of UQ process steps by leveraging exascale calculations

Links

PC2-WP1/2/3/4, PC3-WP2/3

Specifications / Tasks

- ▶ T6.1 Extension of kernel methods to complex inputs/outputs
- ▶ T6.2 global sensitivity analysis (GSA)
- ▶ T6.3 GSA in the presence of uncontrollable stochastic random input
- ▶ T6.4 Multi-scale GSA in code coupling/chaining
- ▶ T6.5 GSA with complex input
- ▶ T6.6 Links between kernel-based sensitivity indices (HSIC, MMD) and total Sobol indices

Exa-MA: Methods and Algorithms at Exascale



WP7: Demonstrate methods and algorithms at Exascale

- ▶ T7.1 Benchmarking on small/mini apps
- ▶ T7.2 Co-design with the CDT and PC5
- ▶ T7.3 Showroom of methods and algorithms
- ▶ T7.4 Training

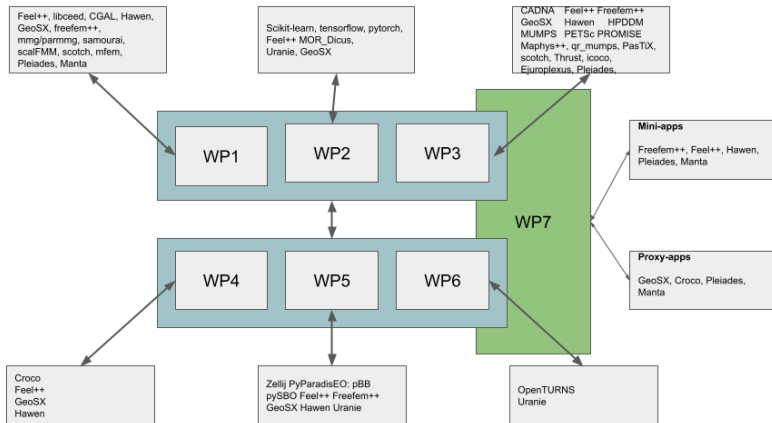
Links

PC2-WP1/2/3/4, PC3-WP2/3 and PC5

Exa-MA: Methods and Algorithms at Exascale



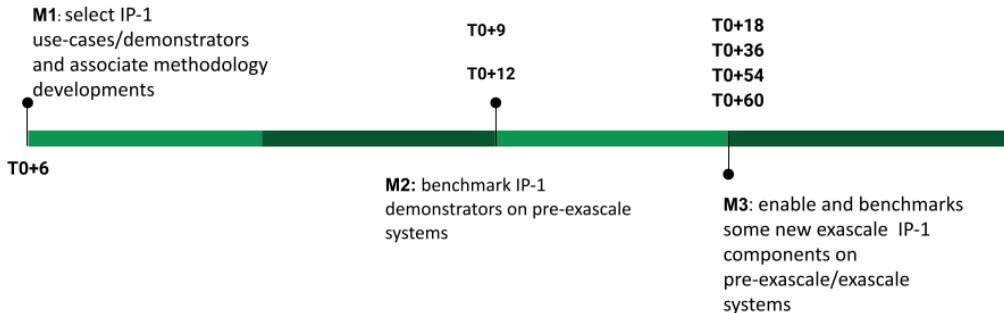
Software - WIP, > Software List



Exa-MA: Methods and Algorithms at Exascale



Agenda



Exa-MA: Methods and Algorithms at Exascale



Needs wrt HPC ressources

Software stack

- ▶ Enable modern C++ (C++ ≥ 17) software, some Fortran and Python software (using wrapping C++ code)
- ▶ Enable complex software stack and possibly CI/CD pipelines

Runs

- ▶ run the benchmarks (mini-apps and proxy-apps) on the HPC platforms, usually "short" runs (minutes to a few hours) however a wide range of code typology
- ▶ run on hybrid hardware (not necessarily at scale)

Exa-MA: Methods and Algorithms at Exascale



Status : Teams

Partners

- ▶ CEA : DES+DAM
- ▶ INRIA : Bordeaux, Côte d'Azur, Grenoble, Lille, Lyon, Paris, Saclay
- ▶ IPP : CMAP
- ▶ Sorbonne Université : LJLL+LIP6
- ▶ UNISTRA : IRMA+Cemosis