Internship Opportunity: Optimizing Linear System Solvers on GPGPU Architectures for ArcaneFEM

Subject: GPU, Parallel computing, Linear Solvers, AMG, HPC, FEM, `C++`

Context: GPGPU[1]-based numerical simulations rely on heterogeneous computing frameworks which is a major driver of the ongoing transition to the exascale computing[2] era. As large-scale parallel numerical simulations increasingly rely on GPU compute power, existing numerical solvers/codes must evolve to fully embrace the massive parallelism offered by modern GPUs. In this context, `ArcaneFEM` [1] — a massively parallel HPC finite element solver — has been steadily maturing to support GPGPU computations for multiple problems of physics such as earthquake modeling (fig. 1-right), linear elasticity, diffusion problems, heat transfer, aerodynamics (fig. 1-left), etc. `ArcaneFEM` is built on top of the `Arcane Framework` [2] — a `C++` development environment for parallel numerical simulation codes — and thus inherits its versatile parallelization capabilities, including support for CPUs, GPUs, threads, and more. `ArcaneFEM` is designed to scale across thousands of CPUs and GPUs, making `ArcaneFEM` a strong candidate for this exascale evolution.

A key computational challenge in any finite element method based numerical simulations is the solution of large, sparse linear systems, it is worth mentioning that linear systems can be as large as containing billions of unknowns, for instance in earthquake simulations. `ArcaneFEM` currently utilizes Algebraic Multigrid (AMG) preconditioned iterative solvers on thousands of GPUs to solve such systems [3]. While this already offers significant speed-ups, there is considerable room for performance tuning, robustness improvements, and better exploitation of future GPU architectures. This internship aims to advance the linear solver capabilities of `ArcaneFEM` in this direction.

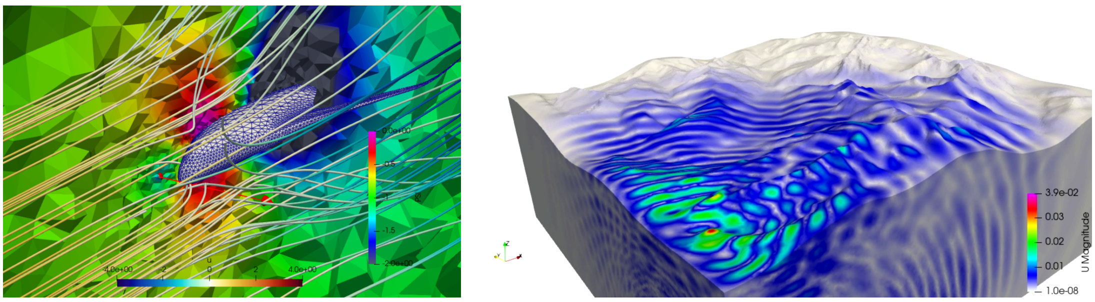

Figure 1: Illustrations of aerodynamics (left) and earthquake (right) numerical simulations obtained with `ArcaneFEM`.

The selected intern will join the SGLS (Service de Génie Logiciel pour la Simulation) at CEA Paris-Saclay, where `ArcaneFEM` is actively developed and the internship will be co-supervised by experts from CEA DAM (La Direction des applications militaires). The student will work in a team of experts in HPC, numerical methods, and code performance, with access to powerful computing clusters. Emphasis will be placed on testing, tuning, and benchmarking solvers on real HPC infrastructures, to ensure the solvers scale effectively and remain robust across a wide range of applications.

[1] GPGPU: General-Purpose computing on Graphics Processing Units — the use of GPUs, traditionally used for rendering graphics, to perform computations typically handled by the CPU in scientific and engineering applications.

[2] Exascale Computing: A class of super-computing systems capable of executing over 10^{18} floating-point operations per second (exaFLOP).

Objectives: The internship will involve the following tasks:

- Improve and tune linear solvers currently used in `ArcaneFEM` on GPGPU.
- Investigate alternative solver kernels or preconditioners optimized for modern GPU architectures.
- Use HPC clusters to run large-scale tests and benchmark solver performance and scalability.
- Contribute to code integration and develop tools to monitor solver behavior.

Requested knowledge:

- Good knowledge of `C++` programming.
- Experience with GPU programming (`CUDA`, `HIP`, or equivalent) is a plus.
- Understanding of linear solvers and iterative methods is a plus.
- Motivation to work with HPC systems and interest in scientific software engineering.

Period: 5 to 6 months (September 2025)

Profile: BAC+3 onwards (master, grande école, L3, or equivalent)

Locality: CEA Paris-Saclay, at (SGLS/LESIM Lab)

To apply: Send your CV, grades, and motivation letter to

- Mohd Afeef BADRI: mohd-afeef.badri@cea.fr
- Gilles GROSPELLIER: gilles.grospellier@cea.fr

References:

[1] M.A. Badri, et. al., (2025). ArcaneFEM: Massively parallel FEM solver on CPU-GPU. GitHub. https://github.com/arcaneframework/arcanefem

[2] G. Grospellier, et. al., (2009). The Arcane development framework. POOSC'09. https://doi.org/10.1145/1595655.1595659

[3] J. Bolz, et. al., (2020). Sparse matrix solvers on the GPU: conjugate gradients and multigrid. ACM Journals. ACM Transactions on Graphics. Vol. 22, No. 3 https://doi.org/10.1145/882262.882364