2SC7692 – Shape optimization and drag reduction in aeronautics

Instructors: Stephane Vialle
Department: DOMINANTE - MATHÉMATIQUES, DATA SCIENCES, DOMINANTE - INFORMATIQUE ET NUMÉRIQUE
Language of instruction: ANGLAIS
Campus: CAMPUS DE PARIS - SACLAY
Workload (HEE): 80
On-site hours (HPE): 48,00

Description
Project topic in partnership with ONERA.

Air traffic is steadily increasing each year to the point that, without improvements in aircraft performance in terms of energy consumption, the share of air transport in greenhouse gas emissions may become unsustainable in the future.

Aircraft consumption can be decreased by either increasing the engine efficiency or by improving the aerodynamic design of the aircraft, e.g. reducing the aircraft weight. Computational tools have been widely used in aeronautics for a long time to help design and optimize systems. For example the shape of a wing can be improved to reduce its drag, lift, or its inner structure can be lightened.

Technical details of the system and methodology:

Optimization methods require successive calculations for different wing geometries including the calculation of *adjoint models*. The computational costs for each step can become prohibitively expensive for high fidelity numerical models.

The only way to reduce computing times such that results can be obtained fast enough to integrate optimization methods into the industrial design cycle is to use parallel computers. In the case of optimization methods like efficient descent methods (such as the gradient method or the Newton method), the different configurations are not known a priori but determined successively by the algorithm. It is therefore necessary to parallelize each calculation of the *primal problem* and then the *adjoint problem*.

The objective of this project is to achieve the parallelization of the most expensive phase of the optimization loop, namely the resolution of large

linear systems resulting from finite element discretization models on large meshes to experiment with different sets of optimization parameters.

For this, the parallelization will be performed, in a message exchange programming environment adapted to the use of very large computers with network computing nodes, by a domain decomposition approach. The global iterative resolution method will be accelerated by solving the local equations in each subdomain. The developed parallel code will be executed and evaluated on parallel machines of CentraleSupélec.

Quarter number
ST7

Prerequisites (in terms of CS courses)
First year courses:

- SG1 common course "Systèmes d'Information et Programmation" (1CC1000)
- ST2 common course "Algorithmique et complexité" (1CC2000)

Courses of the ST:

- ST7 common course "Optimisation" (2CC3000)
- ST7 specific course "Méthodes et algorithmes parallèles pour l'optimisation" (2SC7610)

Others prerequisites:

- Parts of common course "CIP - Convergence, Intégration et Probabilités" (1SL1000)
- Parts of common course "EDP - Equations aux dérivées partielles" (1SL1500)
- Knowledge of linear algebra will also be needed

Syllabus

Main stages of the study:

- Additional information on the subject at the beginning of the first part of the project (presented by ONERA):
    - presentation of fundamental theoretical tools in optimization used in aeronautics
    - presentation of the scientific issues related to the design of aircraft wings, the evaluation of the drag and the estimation of the associated risk when reducing the weight of the materials used.

- Formalization of the problem of representing geometry and structure to characterize the drag of an airplane wing. Use of a fixed optimization method adapted to the problem (method of geometry optimization using the Hadamard representation of the shape derivative). Development of the algorithm on a verification case supported by the analytical solution.
- Identification of the FENICS (OpenSource) code parameters for solving partial differential equations, and their impact on the quality of the flow calculation solution and on the execution time.
- Design and implementation in Python of an optimization code calling FEniCS; optimization code which will then be parallelized by message exchanges based on a domain decomposition approach.
- Test and development of the complete optimization code on a parallel machine of the CentraleSupélec Teaching Data Center, on small and medium problems (short duration of physical simulation, small size of the domain).
- Application to a real case (in terms of the size of the studied area, and geometry):
    - Experiments with larger problems on a larger number of nodes and computing cores (scaling approach),
    - Analysis of the quality of the solution and the performance of the calculations,
    - Optimization of the code to improve the solution quality AND, if needed, the performance of the calculations,
- Estimation of the optimal shape of the wing or flaps to reduce the drag of the flow.
- The study will conclude with the submission of a report and an oral presentation to evaluate: the quality of the proposed solution, the efficiency and extensibility of the algorithm in finding an optimal solution, and the management of the computation resource quota during the project.

Rmk: Different student groups will enforce different optimization methods, evaluated on different examples.

Class components (lecture, labs, etc.)

Part 1 (40HEE):

- Steps 1 and 2: course complements, handling of computing resources, formalization of the problem, introduction to the optimisation method.
- Step 3 and 4: Python numerical implementation of the optimisation algorithm used. Result analysis in order to choose the parameters to optimize and the quality of the solution.

- Step 5: first implementation of the optimisation algorithm on parallel machine, evaluation of the result quality and computation performances on small and medium size problems, and comparison to a reference analytical solution.

- Midterm report (slides and progress summary), and talk about current results and future work (part 2).

Part 2 - *final sprint* (40HEE):

- Step 6: experiments on real problems with different configurations, identification of the sources of performance locks and losses, and (if required) code improvement to push the limits of supportable problem size.
- Step 7: estimate of the optimal shape of the wing or shutters in order to reduce the airplane drag, function of the available computing resources and of the maximum duration of the project.

- Final report (slides and extended abstract) and global talk about the project approach and results.

Grading

This project will be evaluated by a midterm talk at the end of part 1 (40HEE), and by a final talk at the end of part 2 (*final sprint* 40HEE). Talks will be done by the entire team, but will lead to individual marks in case of strongly heterogeneous teams.Each talk evaluation will consider the overall quality of the talk, of the slides and of the progress summary. Each talk mark will be 50% of the total mark.

Resources

Teaching staff:

- **F. Magoules** (CentraleSupélec & MICS), and **S. Vialle** (CentraleSupélec & LISN)
- **S. Claus** and **F.-X. Roux** (ONERA)

Workplace and computing resources:

- Students will work at CentraleSupelec, in a classroom with electrical outlets and reliable wifi Internet access.
- Students will use their laptops to connect to remote PC clusters at the Data Center for Education of CentraleSupelec.
- Final oral exam will take place at CentraleSupelec the last afternoon of the project.

Learning outcomes covered on the course

When finishing the course, the students will be able to:

- Learning Outcome 1 (**AA1**): identify and parametrize an optimisation method adapted when each evaluation/iteration requires many computing resources and computation time,
- Learning Outcome 2 (**AA2**): implement and debug a sequential or parallel Python code on supercomputer (developped from scratch or using libraries), calling distributed C/C++ computing kernels,
- Learning Outcome 3 (**AA3**): deploy intensive computing applications on remote resources,
- Learning Outcome 4 (**AA4**): identify the limitations of the study, function of the available computing resources,
- Learning Outcome 5 (**AA5**): manage a computing resource quota, during an intensive computing campaign.

Description of the skills acquired at the end of the course

- C4: Have a sense of value creation for his company and his customers
- C7: Know how to convince
- C8: Lead a project, a team