

# CIRM Summer School on Domain Decomposition for Optimal Control: An introduction to Domain Decomposition Methods

Victorita Dolean

July 17, 2022

**Resources:** A part of the lecture will follow a few chapters of the book "An introduction to domain decomposition methods: algorithms, theory and parallel implementation", SIAM 2015. This book is freely downloadable together with the Freefem codes used to illustrate the methods: <http://www.victoritadolean.com/p/book.html>. Installation of the open source software Freefem (<https://freefem.org/>) is recommended. (available for all platforms both in binary and compiled version)

## Lecture 1: A basic introduction to DD methods

**Objectives:** This presentation will be kept at a basic level, both continuous and algebraic versions of the methods will be given in their most common variants and the main ingredients of domain decomposition methods will be presented. The content will follow the lines of the chapters 1 and 3 from the domain decomposition book. A short introduction to Freefem software will be given which will allow the students to use quickly the codes illustrating the methods.

**Outcomes:** At the end of this first lecture, students will have a basic understanding of the methods but also of their implementation.

## Lecture 2: Two-level methods and preconditioners

**Objectives:** Domain decomposition methods are meant to be used as parallel solvers and scalability (behaviour independent of the number of subdomains/processors) and robustness with respect to the physical parameters are very important issues. An introduction to coarse spaces and two-level methods for symmetric positive definite (SPD) problems will be given together with the presentation of a few variants of domain decomposition preconditioners (AS, RAS, ORAS, SORAS). The content will follow chapters 4 and 5 from the book, although more recent research results will also be included.

**Outcomes:** Students will be able to understand the use and the impact of the two-level methods both for scalability and robustness (even if at this stage the codes are sequential).