C.E.P.S. : an efficient tool for cardiac electrophysiology simulations

Mehdi Juhoor, Yves Coudière, Nejib Zemzemi

C.E.P.S : an efficient tool for cardiac electrophysiology simulations

MEHDI JUHOOR, YVES COUDIÈRE, NEJIB ZEMZEMI

mehdi.juhoor@inria.fr, yves.coudiere@inria.fr, nejib.zemzemi@inria.fr

PROVIDE A UNIQUE PLATFORM FOR CARDIAC ELECTROPHYSIOLOGY RESEARCH

ABOUT THE PROJECT

Being developed at INRIA Bordeaux, CARMEN team

Main goals:

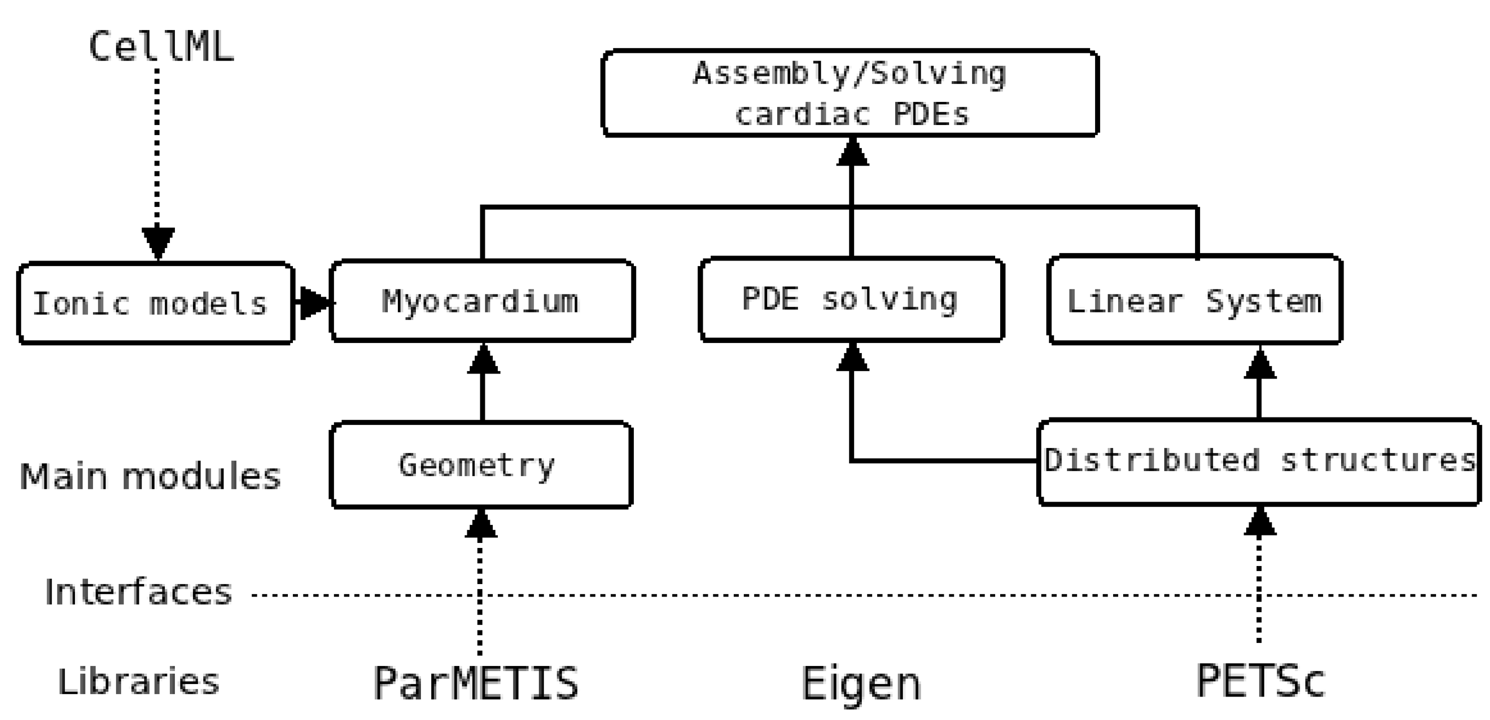
- **Simulating the electrophysiology of the heart** from cell ionic activity to body surface.
- **Modelization of forward problem and solving of inverse problem** in cardiac electrophysiology

CODE DESIGN

- **Highly specialized code**
- **Ready to run on HPC platforms**
- **Enable high-order numerical methods**
- **Easily add new PDE/ODE systems**
- **Make use of efficient and well-known libraries**

GENERAL ARCHITECTURE

MODULES



General view of CEPS modules

DESIGN

C++ code, aiming for:

- genericity
- efficiency
- readability
- portability

Documentation generation via Doxygen

Using alternative libraries is possible through interface implementation

Produced code fully tested with CxxTest and continuous integration through Jenkins

COMPUTATIONAL DOMAINS

MOTIVATION

Compute on major structures of the heart in a single simulation :

- ventricles
- atria
- conduction system network

Express the coupling between these structures to account for the conduction of the electrical impulse.

DISCRETIZATION

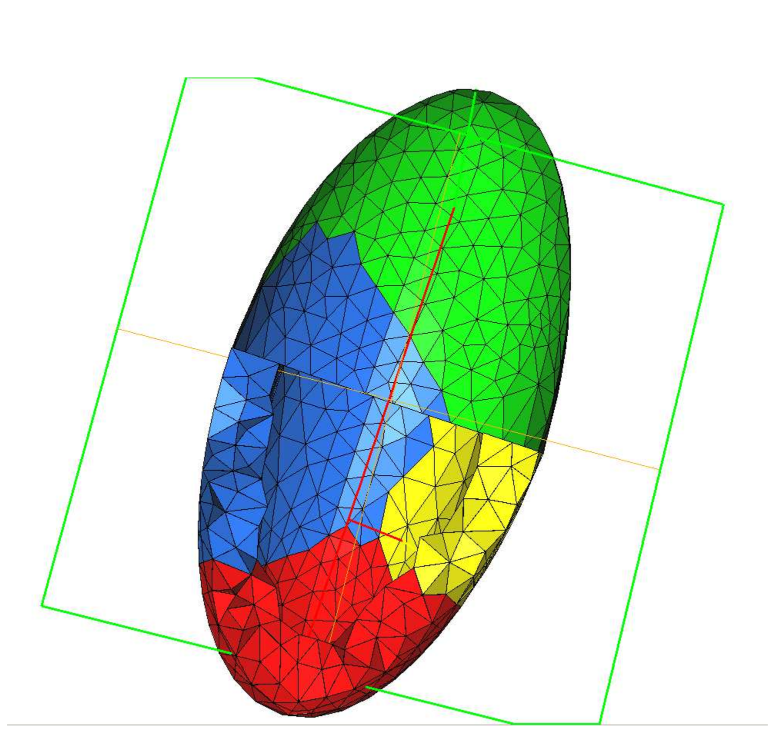
Each structure is represented in 3D space by a mesh

- volumic (ventricles/thorax)
- surfacic (atria)
- cable (fibers of conduction system)

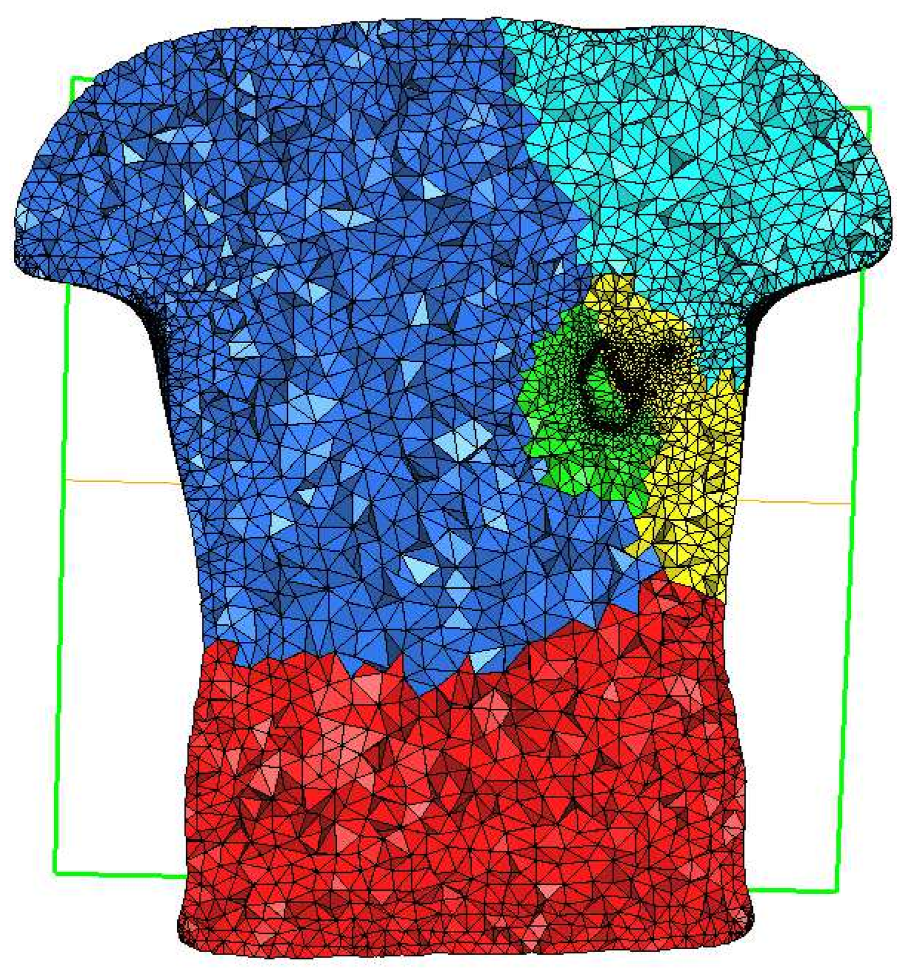
Partitioning of the geometry in order to:

- achieve load balancing
- minimize communications
- reduce computing time

GEOMETRY EXAMPLES



Volumic, surfacic and cable elements in a single partitioned geometry. Each process is assigned a color.



Volumic geometry of heart and torso.

PARALLEL IMPLEMENTATION

COMPUTATIONAL CHALLENGES

Accurate models are increasingly complex ⇒ expensive computations

Computing time increases with:

- problem size
- fine space/time discretization
- accurate numerical methods
- electrophysiological ionic models

IMPLEMENTATION

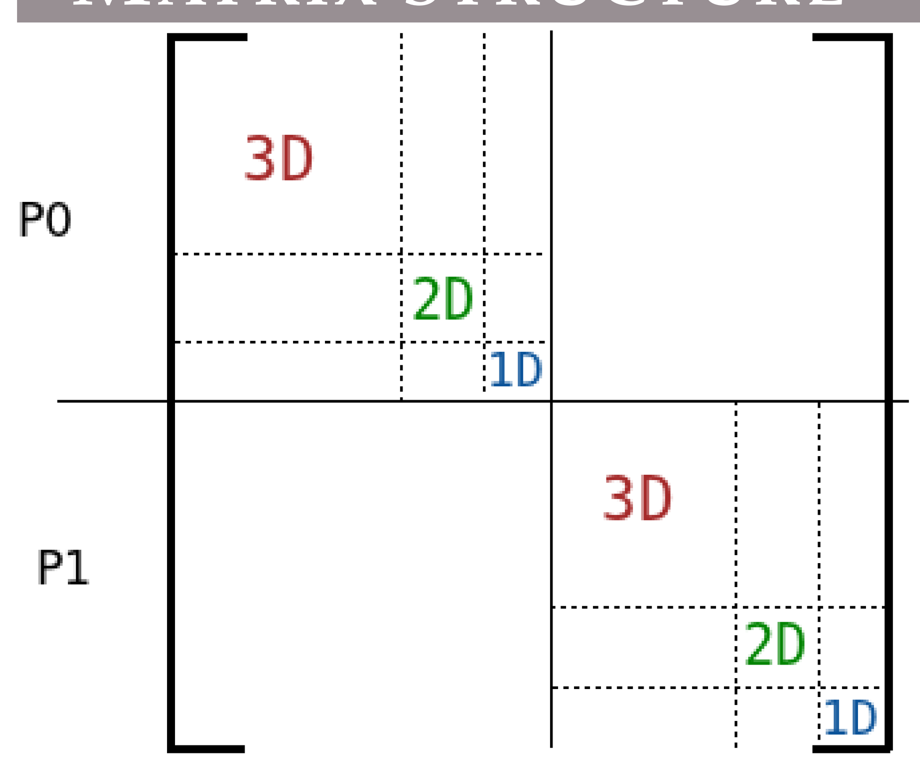
Parallel computing design of each component of the code

- **Mesh partitioning**: ParMETIS[2]
- **Linear system solving**: PETSc[1]
- **I/O**: HDF5[3]

High level of abstraction, transparent way of manipulating distributed data structures.

PROBLEM DISCRETIZATION

MATRIX STRUCTURE



Discretization of cardiac PDEs lead to linear systems in the form $Ax = b$

This figure shows the **parallel structure** of distributed matrix A for **multi-dimensional geometries**

Coupling between elements of different dimension will be expressed in this matrix

NEXT STEPS

SHORT-TERM

Implementation of:

- ionic models
- multi-dimensional coupling
- assembly/resolution of monodomain/bidomain equations

LONG-TERM

Integration with other LIRYC tools

- CARP (Edward J. Vigmond)
- CardioViz(Imaging)
- ...

REFERENCES

[1] Satish BALAY et al. *PETSc Web page*. http://www.mcs.anl.gov/petsc. 2013.

[2] George KARYPIS. *ParMETIS Web page*. http://glaros.dtc.umn.edu/gkhome/metis/parmetis. 2013.

[3] THE HDF GROUP. *Hierarchical data format version 5*. 2000-2010. URL: `http://www.hdfgroup.org/HDF5`.