Data driven Computational Mechanics at EXascale

Data driven Computational Mechanics at EXascale

Work program topic: EuroHPC-01-2019
Type of action: Research and Innovation Action (RIA)

PERFORMANCE EVALUATION REPORT

DELIVERABLE D8.1

Version No 0.2

http://www.dcomex.eu/
This project has received funding from the European High-Performance Computing Joint Undertaking Joint Undertaking ('the JU'), under Grant Agreement No 956201

DOCUMENT SUMMARY INFORMATION

Project Title	**Data driven Computational Mechanics at EXascale**
Project Acronym	DCoMEX
Project No:	956201
Call Identifier:	EuroHPC-01-2019
Project Start Date	01/04/2021
Related work package	WP 8
Related task(s)	Task 8.1, 8.2
Lead Organisation	NTUA
Submission date	9/10/2024
Re-submission date	
Dissemination Level	PU

Quality Control:

	Who	Affiliation	Date
Checked by internal reviewer	Vissarion Papadopoulos	NTUA	27/09/2024
	George Stavroulakis	NTUA	27/09/2024
Checked by WP Leader	Vissarion Papadopoulos	NTUA	27/09/2024
Checked by Project Coordinator	Vissarion Papadopoulos	NTUA	27/09/2024

Document Change History:

Version	Date	Author (s)	Affiliation	Comment
0.2	30.10.2024	Dimitris Dellis George Stavroulakis Konstantinos Atzarakis	GRNET NTUA	Version 2

Table of Contents

1. Executive Summary

This report provides a performance report for the DCoMEX project and specifically for the DCoMEX-BIO framework. Here, we define the testing and benchmarking procedures that will be utilised for this report. In particular, we define:

- The design and architecture of the test cases considered
- The actual results in terms of computational time, speedup and network efficiency

2. Application benchmarks

Currently, the DCoMEX framework is a set of 2 application codes relying on MSolve. In the sections below, we describe the benchmark applications, the benchmark problems and the datasets.

MSolve

MSolve, is a general-purpose computational mechanics solution platform with multiphysics and multiscale capabilities, developed at NTUA, that exploits combined multi-core central processing units (CPU) and graphics processing units (GPU).

1. Relevant Links
- Github: https://github.com/mgroupntua
- Documentation: https://mgroupntua.github.io/
2. Development Status
- TRL 6 - Under refactoring, most unit tests work.
3. Hardware Support
- CPU Intensive computation (Dense Linear Algebra mostly) using MKL
- Local Parallelism - It employs local parallel sampling through fork/join (multiprocessing).
4. Dependencies
- (Optional) SuiteSparse - For linear algebra module
- (Optional) Intel MKL - For linear algebra module
- Compiler: Roslyn, support for C# 7.0 and higher.
5. Platform Support
- Linux: Compiling
- MacOS: Compiling
- Windows: Compiling

2.1 DCoMEX-BIO

DCoMEX-BIO is a multiphysics modeling framework built on MSolve, designed to efficiently analyze and optimize immunotherapy strategies, and as a particular use case the administration of anti-PD-L1 antibody treatments. This framework integrates experimental data with advanced computational models to simulate tumor-immune system interactions, focusing on identifying optimal time sequences for immunotherapy to enhance treatment outcomes. The approach addresses the inherent complexity and uncertainty of the tumor microenvironment, where interactions between cancer cells and immune system agents play a crucial role in determining the success of immunotherapy. By leveraging advanced simulation techniques, DCoMEX-BIO aims to provide insights into how different treatment protocols can improve therapeutic efficacy under various conditions.

The models at the heart of this framework use continuum mechanics to simulate the interactions within the tumor and its surrounding healthy tissue. A biphasic theory of soft tissues is employed to model the tumor/host tissue system, capturing both the hyperelastic properties of the solid tumor and the porous flow of interstitial fluid within the tissue. Tumor growth is closely tied to the concentration of cancer cells, which proliferate in response to oxygen availability and are attacked by various immune agents such as effector CD8+ T-cells. These interactions are modeled through a coupled set of partial differential equations (PDEs), where the concentration of cancer cells and immune agents are governed by advection-diffusion-reaction equations. Additionally, tumor growth is described by ordinary differential equations (ODEs), solved pointwise at each integration point within the porous medium of the tumor region.

Two primary benchmarks have been developed within this framework to explore and validate different aspects of tumor-immune dynamics and treatment response. The first is a simplified 5-equation model, which captures the proliferation of cancer cells of a soft biphasic tissue, in the presence of oxygen. The second, more comprehensive 12-equation model, delves into greater detail, incorporating the response of immune system agents and their interactions with each other and the injected anti-PDL1 anti bodies, within a three-dimensional tumor-host tissue domain. Both benchmarks leverage advanced numerical methods to solve the coupled system of equations, and both models have been validated against commercial software such as COMSOL, ensuring their accuracy in predicting tumor growth and response to anti-PD-L1 injections.

To overcome the immense computational cost associated with the Bayesian update of such a model, efficient neural network-based surrogates are developed by use of the AI-Solve library, allowing for the calculation of the time evolution of the total volume of the tumor region. This volume value serves as an indicator of the efficiency of the immunotherapy treatment, while the parameters of the model defined stochastically due to uncertainty in their exact values. Hence, the creation of a large dataset is necessary to capture the variability and complex dynamics of tumor growth and immune response across different treatment scenarios. This dataset will include detailed simulations that reflect variations in both biological and therapeutic factors, providing the essential data required for training the surrogate model.

The dataset will be leveraged to build a reliable surrogate model capable of replicating the key outcomes of the full multiphysics simulation but with drastically reduced computational demands. This surrogate will maintain accuracy in predicting the tumor’s behavior and immune response, making it possible to quickly evaluate various therapeutic strategies without running resource-intensive simulations.

Ultimately, this surrogate model will facilitate the update of stochastic parameters in the multiphysics model, in the context of a Bayesian inference procedure, resulting in a powerful AI-assisted tool that can accurately predict tumor evolution. This tool will be used to fine-tune treatment protocols, optimizing the timing and dosage of interventions to improve immunotherapy outcomes and personalize treatment strategies effectively.

2.1.1 Studied cases, datasets and corresponding models

5-Equation Model Benchmark

The 5-equation model benchmark is a testing prototype, that employs a simplified approach to tumor-soft host tissue interactions and captures the essential dynamics of cancer cell proliferation, oxygen consumption and growth in a hyperelastic porous medium. This model is based on the DCoMEX-BIO prototype (D7.2) and focuses on core interactions while reducing computational demands.

The mathematical system includes four partial differential equations (PDEs), modeling hyperelastic porous medium, immune cell proliferation, and oxygen transport and an ODE solved point-wise in

space modeling tumor growth. The model relies on Darcy's law for interstitial fluid flow and simple advection-diffusion-reaction formulations to capture the transport and interaction cancer cells and oxygen.
Technically, the domain for this benchmark represents a single spherical tumor in a host tissue modeled using simple symmetry boundary conditions. The validation procedure followed for this model is given in detail in report D7.2.
Main purpose of the dataset that was created using this model, is its utilization in the creation of a surrogate model that enables the prediction of both the spatial and temporal evolution of the simulated fields. It enabled the assessment of the proposed methodology for the creation of surrogate models of this kind, described in detail in deliverable report D7.5.

12-Equation Application

The 12-equation model represents a more advanced and detailed simulation, designed to closely replicate the response of the immune system and its interaction with anti-PDL1 therapy considering tumor growth in a hyperelastic porous medium. This model extends the 5-equation version by incorporating additional immune system agents, providing a deeper exploration of tumor-immune system dynamics. The total assembly of equations is comprised of the porous hyperelastic medium equations (2 pdes), cancer cell and oxygen concentration (2 pdes), growth coefficient (point-wise solved ODEs), anti-PDL1 anti body concentration and 6 equations corresponding to the considered agents of the immune system { T_E, *IAPC, APC, c,* A_g, *In}* that are described in detail in report D7.5.
Main purpose of the dataset that was created using this model, is its utilization in the creation of a surrogate model focused on the prediction of key indicators of the efficiency of the immunotherapy, such as the volume of the tumor region at discrete time points. The proposed methodology for the creation of these models is given in D7.5 as well.

2.1.2 Test cases

Regarding the performance evaluation three test cases were considered differing in the mesh discretization and the model choice.

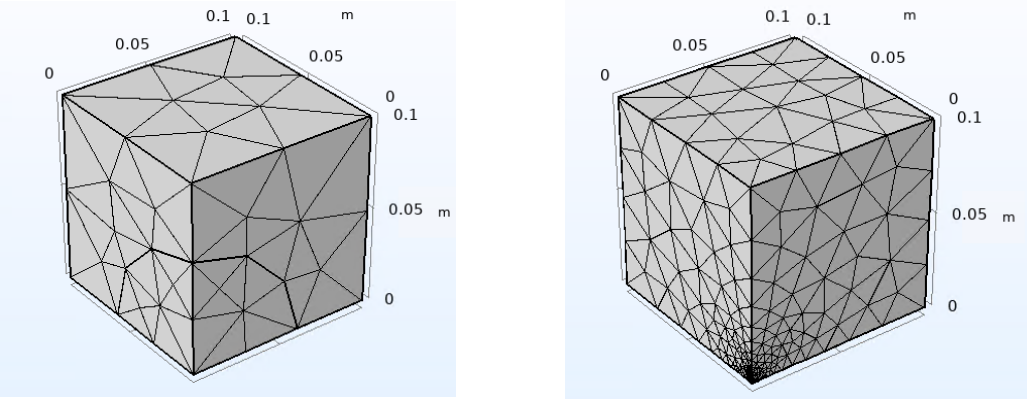

Figure 1 *: Computational mesh domains for test cases: "5equations_small"and "5equations_medium"*

"5equations_small", "5equations_medium"

This benchmark simulates a 3D host tissue domain with dimensions of 0.2 m × 0.2 m × 0.2 m, within which a spherical tumor region is placed at the center. A tumor region of a radius: 0.05 m is considered in the "5equations_small" benchmark and a radius of 0.0004 m is considered for the

"5equations_medium" case. The complexity of the model necessitates the use of symmetry boundary conditions, reducing computational requirements by solving only one-eighth of the domain. On these symmetry planes, normal displacements are fixed, while tangential slip degrees of freedom are permitted.
A time-dependent solver is applied, utilizing fine time-stepping in the early phases to capture rapid changes, followed by progressively larger time steps as the simulation progresses. The benchmark covers a total simulation time of 6 days, or 518400 seconds.
The computational mesh includes approximately 78/733 vertices respectively, with a total of 329/3702 degrees of freedom and an additional 243/2815 internal degrees of freedom. The solver employs a Generalized Alpha Solver, with second-order accuracy for the porous medium mechanics, for the growth ODE and advection-diffusion-reaction equations. Newton-Raphson iteration is used to solve nonlinearities within the system. Validation of this model has been performed through comparisons with COMSOL software, ensuring that the simulated tumor progression and immune response match observed data.
The key technical parameters and initialization settings (e.g., initial oxygen and cancer cell concentration, initial stretch ratio etc.) are provided in report D7.2.

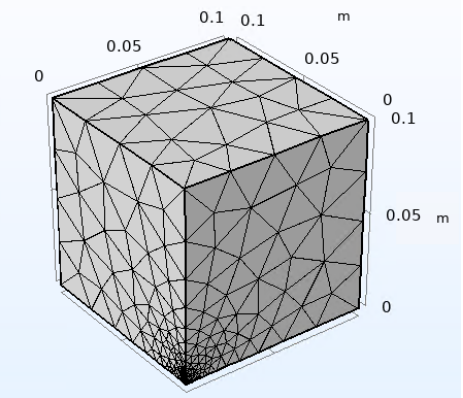

***Figure 2** : Computational mesh domain for test case: "12equations_small""*

"12equations_small"

This benchmark simulates a 3D host tissue domain with dimensions of 0.2 m × 0.2 m × 0.2 m as well, within which a spherical tumor region (radius: 0.003 m) is placed at the center. Symmetry boundary conditions are used once again, reducing computational requirements by solving only one-eighth of the domain. On these symmetry planes, normal displacements are fixed, while tangential slip degrees of freedom are permitted. For the concentration equations of the immune agents, no mass flux is allowed across these planes.
A time-dependent solver is applied, utilizing fine time-stepping in the early phases to capture rapid changes, followed by progressively larger time steps as the simulation progresses. The benchmark covers a total simulation time of 14 days, or 1,209,600 seconds, to study the tumor's evolution over a typical experimental period. Anti-PDL1 antibody injections are modeled as transient events occurring at different intervals (e.g., days 7, 10, and 13), and the injection profile includes second-order continuous derivatives with an exponential decay to simulate realistic drug release.
The computational mesh includes approximately 617 vertices, with a total of 6952 degrees of freedom and an additional 2384 internal degrees of freedom. The solver employs a Generalized Alpha Solver, with second-order accuracy for the porous medium mechanics and first-order accuracy for the growth

ODE and advection-diffusion-reaction equations. Newton-Raphson iteration is used to solve nonlinearities within the system. Validation of this model has been performed through comparisons with COMSOL software, ensuring that the simulated tumor progression and immune response match observed data.
The key technical parameters and initialization settings (e.g., initial cancer cell concentration, immune agent levels, and stretch ratio) are provided in D7.5 report.

3. Benchmark results

The following benchmarks have been run on MeluXina, on the cluster nodes, featuring AMD EPYC 7H12 64-Core Processors, 512GB of RAM and an InfiniBand (IB) HDR 200Gb/s high-speed fabric for node interconnection.

3.1 DCoMEX-BIO

Problem: 5equations_small

Nodes	Time (min)	Speedup (x)	Efficiency (%)	Timesteps
10	148.0	1.0	100	700
100	15.2	9.7	97	690
1000	1.6	92.5	92	685

Problem: 5equations_medium

Nodes	Time (min)	Speedup (x)	Efficiency (%)	Timesteps
10	1702.1	1.0	100	730
100	173.4	9.8	99	730
1000	18.9	90.1	98	735

Problem: 12equations_small

Nodes	Time (min)	Speedup (x)	Efficiency (%)	Timesteps
10	40.3	1.0	100	1504
100	4.2	9.6	97	1500
1000	0.4	93.6	93	1502

4. Summary and Conclusions

In this report, the performance of DCoMEX-BIO was examined. In all test cases, scalability was near the optimum and all available computational resources were utilized in an optimum manner.