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)

Protocols for immunotherapy

DELIVERABLE D7.5

Version No 1

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 3
Related task(s)	Task 3.1, 3.2, 3.3, 3.7
Lead Organisation	NTUA
Submission date	27/09/2024
Re-submission date	
Dissemination Level	PU

Quality Control:

	Who	Affiliation	Date
Checked by internal reviewer	Vissarion Papadopoulos	NTUA	21.09.2024
	George Stavroulakis	NTUA	21.09.2024
Checked by WP Leader	T. Stylianopoulos	UoCY	21.09.2024
Checked by Project Coordinator	Vissarion Papadopoulos	NTUA	21.09.2024

Document Change History:

Version	Date	Author (s)	Affiliation	Comment
1.0	21.09.2024	Gerasimos Sotiropoulos	NTUA	

Contents

1. Description

Main outcome of the DCoMEX-BIO application is the development of an Ai-assisted Multiphysics simulation-driven optimization framework for identifying optimal treatment strategies and protocols of immunotherapy to improve its performance outcomes, under uncertainty. Specifically, in the application under study, the focus is on finding the optimal time sequence of anti-PDL1 antibody injections, in a simulation driven approach that considers experimental data. Regarding the tumor microenvironment modeling, a continuum mechanics model is used that models the tumor/host tissue system according to the biphasic theory of soft tissues. A hyperelastic model, (a neo Hookean material) is considered for the solid skeleton and the porous flow of the interstitial fluid is described by Darcy's law and a continuity equation accounting for the fluid flux entering from the blood vessels minus the fluid flux exiting through lymphatic vessels. The tumor growth is dependent on the concentration of cancer cells, that proliferate due to oxygen concentration, and are killed by innate cells, immature antigen presenting cell and effector CD8+ T-Cells and interact with the antibodies that may be injected in the tissue. The concentration of these agents is model by advection-diffusion-reaction equations that are solved in a coupled manner with the porous-hyperelastic-growth model (PHG model). The time evolution of the growth coefficient of the tumor region is described by ODEs that are solved point wise at each integration point of the porous medium at the tumor region. The total system of equations to be solved is comprised of 11 pdes and 1 point-wise distributed in space ODE.

The purely physics-based model was implemented in MSolve. The MGroup.DrugDeliveryModel application builds upon the MSolve finite element library that was further extended to support it. The original model of 5 equations described in the DCoMEX-BIO prototype report (D7.2) was refined to account for an arbitrary system of equations to model the immune system agents, solved in a fully coupled manner with the PHG model. It is again validated against widely used commercial software for benchmark problems like the common application setup of anti-PDL1 immunotherapy.

To capture the complex response of the immune system, and multi-agent reactions that take place, and to assist the convergence of the solution algorithm a very fine temporal (Generalized-a algorithm) and spatial discretization is necessary. 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, that allow for the calculation of the time evolution of the total volume of the tumor region, that is considered as an indicator of the efficiency of the immunotherapy treatment, in terms of parameters of the model that are defined stochastically because of the uncertainty in determining their value. Upon executing the Bayesian update of those parameters, the inferred model is once again used to create another surrogate model that captures the effect of the different time sequence strategies of the immunotherapy injections. The latter is used in a stochastic optimization framework.

This document is structured as follows: Section 2 describes the mathematical model of the Tumor microenvironment, that was implemented in MSolve, that accounts for the interaction of agents of the Immune system and includes the immunotherapy.

Section 3 presents the technical improvements, i.e. object-oriented based design, made upon the original DCoMEX-BIO prototype codebase and the description of the specialized version of the software to account for specific immune system agents and their corresponding equations. Section 4 presents the benchmarks used during the validation process of the specialized version, including extended comparisons of the results of the two software modules.

Section 4 contains a description of the experimental data to be used in the update of the computational model. Section 5 describes the surrogate models that were created in each stage of the development of the model. Section 6 is devoted to the illustration of the Bayesian inference procedure for the update of the tumor growth model parameters based on experimental data. The stochastic definition of the chosen parameters is presented, and the finalized posterior distributions of the model parameters are provided as well.

Finally, Section 7 presents a stochastic optimization framework for the design of an optimal aPDL1 optimization framework where the time-sequence protocol of the injections is considered as an optimization variable.

2. Description of the tumor microenvironment mathematical model

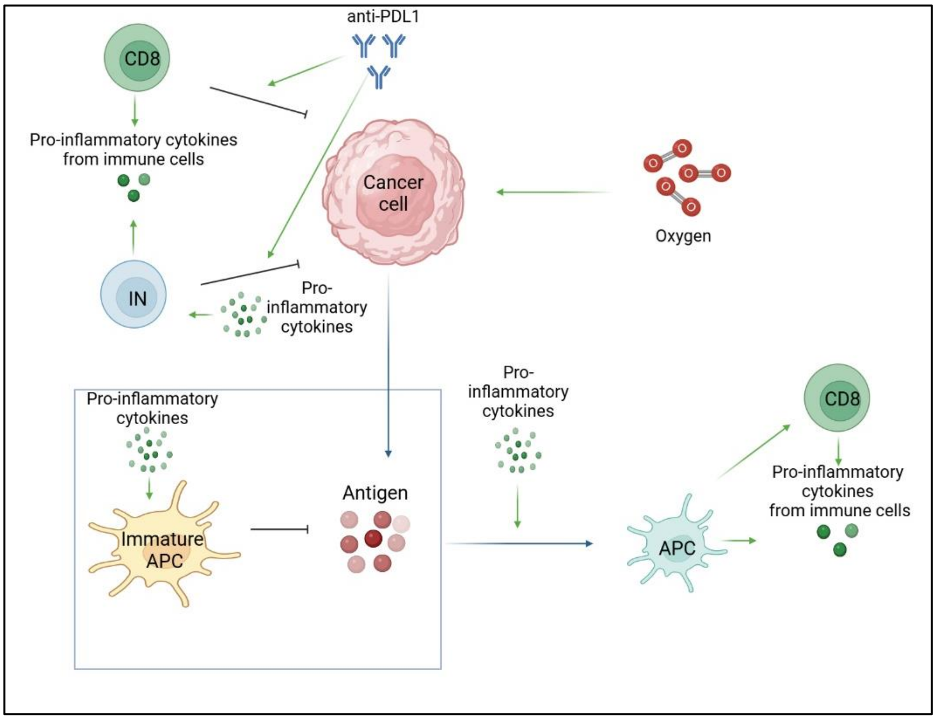
anti-PDL1
CD8
Pro-inflammatory cytokines from immune cells
Cancer cell
Oxygen
IN
Pro-inflammatory cytokines
Pro-inflammatory cytokines
Immature APC
Antigen
Pro-inflammatory cytokines
CD8
Pro-inflammatory cytokines from immune cells
APC

Figure 1*:Immune cell vs cancer cell interactions*

Cancer cell proliferation

The mathematical model accounts for the growth of a spherical tumor surrounded by normal tissue. To calculate the growth rate of the tumor we took into account proliferation of Cancer cells[1–5]. To calculate the growth stretch ratio λ_g we used the expression

$$\frac{d\lambda_g}{dt} = \frac{1}{3}\frac{R_{\overline{T}}}{\overline{T}_0}\lambda_g, \tag{1}$$

The concentration of cancer cells is expressed as:

$$\frac{\partial\overline{T}}{\partial t} + \mathbf{\nabla}\cdot\left(-D_{\overline{T}}\mathbf{\nabla}\overline{\mathbf{T}} + \mathbf{v}^s\overline{T}\right) = R_{\overline{T}} = k_1\left(\frac{c_{ox}}{K_2 + c_{ox}}\right)\overline{T} - C_{aPDL1}(n_{In}In + n_{In}IAPC + n_{ad}T^E)\overline{T} \tag{2}$$

Where the first right-hand side term describes the proliferation of cancer cells due to oxygen and the last term describes the killing of cancer cells by innate cells, immature antigen presenting cells and effector CD8+ T cells.

Enhancing the killing of cancer cells by immune cells with anti-PDL1 antibody

$$C_{aPDL1} = \frac{K_T + k_{on}\overline{T}\,aPDL1}{K_T} \tag{3}$$

Where $\mathrm{C_{aPDL1}}$ describes the increase of cancer cells killing by the immune cells due to the administration of anti-PDL1 antibody. $\mathrm{K_T}$ is a parameter that describes the effectiveness of the treatment. $\mathrm{k_{on}}$ is binding rate constant of anti-PDL1 to the tumor cells $\overline{\mathrm{T}}$

Biphasic formulation of the tumor's mechanical behavior

Tumor growth is modelled based on principles from continuum mechanics and particularly the multiplicative decomposition of the deformation gradient tensor (**F**). The kinematics of the tumor are decomposed into two components, the growth component ($\mathbf{F}_g$) that accounts for the growth of the tumor and the elastic component ($\mathbf{F}_e$) that accounts for mechanical interactions of the tumor with the surrounding normal tissue [23, 24]:

$$\mathbf{F} = \mathbf{F}_e \mathbf{F}_g, \tag{4}$$

The growth component is set to be homogenous and isotropic [1,2,6]

$$\mathbf{F}_g = \lambda_g \mathbf{I}, \tag{5}$$

where λ_g is defined in Eq. (6). The elastic component $\mathbf{F}_e$ of the deformation gradient tensor is determined from Eq. (9) as

$$\mathbf{F}_e = \mathbf{F}\, \mathbf{F_g}^{-1}. \tag{6}$$

The interstitial fluid velocity $\mathbf{v}^f$ depends on the interstitial hydraulic conductivity k_{th} and the interstitial fluid pressure gradient, given by Darcy's law [7]:

$$\mathbf{v^f} = -\mathrm{k_{th}} \boldsymbol{\nabla} \mathbf{p_i} + \mathrm{v^s} \tag{7}$$

Combining Darcy's law with the continuity equation ($\nabla \cdot \mathrm{v^f} = \mathrm{Q}$) yields the steady-state fluid transport model [7,8]:

$$-\mathrm{k_{th}} \nabla^2 \mathrm{p_i} = \mathrm{Q} - \nabla \cdot (\mathbf{v}^{\mathrm{s}}) \tag{8}$$

where Q denotes the fluid flux entering from the blood vessels into the tumor or the surrounding normal tissue minus the fluid flux exiting through lymphatic vessels [9]:

$$\mathrm{Q} = \mathrm{L_p S_v (p_v - p_i) - L_{pl} S_{vl} (p_i - p_l)} \tag{9}$$

where L_{pl}, S_{vl} and p_l are the corresponding quantities for lymphatic vessels, and p_i is the interstitial fluid pressure.

According to the biphasic theory for soft tissues [10], the total stress tensor $\boldsymbol{\sigma}_{tot}$ is the sum of the fluid phase stress tensor $\boldsymbol{\sigma}^f = -p_i\mathbf{I}$ and the solid phase stress tensor $\boldsymbol{\sigma}^s$. As a result, the stress balance is written as:

$$\nabla \cdot \boldsymbol{\sigma}_{\mathbf{tot}} = 0 \Rightarrow \nabla \cdot (\boldsymbol{\sigma}^{\mathbf{s}} - \mathrm{p_i} \mathbf{I}) = 0, \tag{10}$$

$$\mathrm{v^s{=}du/dt} \tag{11}$$

where the stress tensor of the solid phase $\boldsymbol{\sigma}^{s}$ is given by [11]:

$$\boldsymbol{\sigma}^{\mathrm{s}} = \mathrm{J}_{\mathrm{e}}^{-1}\mathbf{F}_{\mathrm{e}}\frac{\partial \mathrm{W}}{\partial \mathbf{F}_{\mathrm{e}}^{\mathrm{T}}}, \tag{12}$$

$$\mathrm{W} = \frac{\mu(-3+\mathrm{I}_1)}{2} + \frac{\mathrm{K}(-1+\mathrm{J}_{\mathrm{e}})^2}{2}, \tag{13}$$

Where W is the strain energy density function, I_1 the first invariant of the right Cauchy-Green deformation tensor and J_e is the determinant of $\mathbf{F_e}$.

Oxygen transport equation

A convection-diffusion-reaction type equation is employed for the calculation of the rate of change of oxygen in the tumor. The reaction term is related to the oxygen transferred from the vessels to the tumor, minus the amount of oxygen consumed by cells [1,2], i.e.,

$$\frac{\partial \mathrm{c}_{\mathrm{ox}}}{\partial \mathrm{t}} + \nabla\cdot\left(\mathrm{c}_{\mathrm{ox}}\mathbf{v}^{\mathrm{f}}\right) = \mathrm{D}_{\mathrm{ox}}\nabla^2\mathrm{c}_{\mathrm{ox}} - \frac{\mathrm{A}_{\mathrm{ox}}\mathrm{c}_{\mathrm{ox}}}{\mathrm{c}_{\mathrm{ox}}+\mathrm{k}_{\mathrm{ox}}}\frac{\overline{\mathrm{T}}}{\mathrm{T}_0} + \mathrm{P}_{\mathrm{er}}\mathrm{S}_{\mathrm{V}}(\mathrm{C}_{\mathrm{iox}} - \mathrm{c}_{\mathrm{ox}}), \tag{14}$$

where c_{ox} is the oxygen concentration, D_{ox} is the diffusion coefficient of oxygen in the interstitial space, A_{ox} and k_{ox} are oxygen uptake parameters, P_{er} is the vascular permeability of oxygen that describes diffusion across the tumor vessel wall and C_{iox} is the oxygen concentration in the vessels.

Pro-inflammatory cytokines from immune cells

The pro inflammatory cytokines can be transported by convection and diffusion:

$$\frac{\partial \mathrm{c}}{\partial \mathrm{t}} + \boldsymbol{\nabla}\cdot\left(-\mathrm{D}_{\mathrm{c}}\boldsymbol{\nabla}\mathbf{c} + \mathbf{v}^{\mathrm{f}}\mathrm{c}\right) = \mathrm{k}_{\mathrm{In}}\mathrm{In} + \mathrm{k}_{\mathrm{T^E}}\mathrm{T}^{\mathrm{E}} + \mathrm{k}_{\mathrm{APC}}\mathrm{APC} - \delta_{\mathrm{c}}\mathrm{c} \tag{15}$$

Where the right-hand side terms describe the production of pro inflammatory cytokines by innate immune cells, effector CD8+ and antigen presenting cells. The last term describes the degradation of cytokines.

Immature antigen presenting cells

The immature antigen presenting cells are expressed as:

$$\begin{gathered}\frac{\partial \mathrm{IAPC}}{\partial \mathrm{t}} + \boldsymbol{\nabla}\cdot(-\mathrm{D}_{\mathrm{IAPC}}\boldsymbol{\nabla}\mathbf{IAPC} + \mathbf{v}^{\mathbf{s}}\mathrm{IAPC}) = \lambda_{\mathrm{IAPC}}\left(\frac{\mathrm{c}}{\mathrm{K}_{\mathrm{cAPC}} + \mathrm{c}}\right) - \delta_{\mathrm{IAPC}}\mathrm{IAPC} \\ -\chi_{\mathrm{APC}}\left(\frac{\mathrm{c}}{\mathrm{K}_{\mathrm{cAPC}} + \mathrm{c}}\right)C_{aPDL1}\mathrm{n}_{\mathrm{In}}\mathrm{IAPC}\overline{\mathrm{T}} - \chi_{\mathrm{APC}}\left(\frac{\mathrm{c}}{\mathrm{K}_{\mathrm{cAPC}} + \mathrm{c}}\right)\mathrm{n}_{\mathrm{A_g}}\mathrm{IAPCA_g}\end{gathered} \tag{16}$$

Where the first right-hand term describes the source of immature antigen presenting cells, the second term the degradation and the last term the reduction due to activation. The last two terms describe the activation to antigen presenting cells. The depends on the pro-inflammatory cytokines and the interaction of immature antigen presenting cells with the tumor cells and antigen.

Antigen presenting cells

The antigen presenting cells are expressed as:

$$\frac{\partial \mathrm{APC}}{\partial \mathrm{t}} + \boldsymbol{\nabla}\cdot\left(-\mathrm{D}_{\mathrm{APC}}\boldsymbol{\nabla}\mathbf{APC} + \mathrm{v}^{\mathrm{s}}\mathrm{APC}\right) = \chi_{\mathrm{APC}}\left(\frac{\mathrm{c}}{\mathrm{K}_{\mathrm{cAPC}}+\mathrm{c}}\right) C_{aPDL1}\mathrm{n}_{\mathrm{In}}\mathrm{IAPC}\overline{\mathrm{T}} + \chi_{\mathrm{APC}}\left(\frac{\mathrm{c}}{\mathrm{K}_{\mathrm{cAPC}}+\mathrm{c}}\right)\mathrm{n}_{\mathrm{A_g}}\mathrm{IAPCA_g} - \delta_{\mathrm{APC}}\mathrm{APC}$$

Where the first two right-hand term describes the increase of antigen presenting cells due to the activation from immature antigen presenting cells and the last term is a degradation term.

Effector CD8+ T cells

The effector CD8+ T cells are expressed as:

$$\frac{\partial \mathrm{T^E}}{\partial \mathrm{t}} + \boldsymbol{\nabla}\cdot\left(-\mathrm{D}_{\mathrm{T^E}}\boldsymbol{\nabla}\mathbf{T^E} + \mathbf{v}^{\mathbf{s}}\mathrm{T^E}\right) = \mathrm{m}_{\mathrm{APC}}\mathrm{APC} - \delta_{\mathrm{T^E}}\mathrm{T^E} \tag{17}$$

Where the first term describes the source of effector CD8+ T cells which is assumed to be analog to the concentration of antigen presenting cells responsible for the activation of CD8+ T cells in the lymph nodes. The last term is the degradation of CD8+ T cells.

Innate cells

The innate immune cells that induce cytolysis are expressed as:

$$\frac{\partial \mathrm{In}}{\partial \mathrm{t}} + \boldsymbol{\nabla}\cdot\left(-\mathrm{D}_{\mathrm{In}}\boldsymbol{\nabla}\mathbf{In} + \mathbf{v}^{\mathbf{s}}\mathrm{In}\right) = \lambda_{\mathrm{In}}\left(\frac{\mathrm{c}}{\mathrm{K}_{In}+\mathrm{c}}\right) - \delta_{\mathrm{In}}\mathrm{In} \tag{18}$$

Where the first term describes production of innate cells which depends on the concentration of pro-inflammatory cytokines.

Antigen

The concentration of cancer cells is expressed as:

$$\frac{\partial \mathrm{A_g}}{\partial \mathrm{t}} + \boldsymbol{\nabla}\cdot\left(-\mathrm{D}_{\mathrm{A_g}}\boldsymbol{\nabla}\mathbf{A_g} + \mathbf{v}^{\mathbf{f}}\mathrm{A_g}\right) = C_{aPDL1}\left(\mathrm{n}_{\mathrm{In}}\mathrm{In} + \mathrm{n}_{\mathrm{ad}}\mathrm{T^E}\right)\overline{\mathrm{T}} - \mathrm{n}_{\mathrm{A_g}}\mathrm{IAPCA_g} \tag{19}$$

Where the first right-hand side term describes the production of antigen induced by the cytolytic effect of innate cells and effector CD8+ T cells. The last term describes the antigen uptake by the immature antigen presenting cells.

Immunotherapy, immune checkpoint blockade with anti-PDL1 antibody

The equations for delivery of immunotherapy have the form:

$$\begin{aligned}\frac{\partial \mathrm{aPDL1}}{\partial \mathrm{t}} &+ \nabla\cdot\left(\mathrm{aPDL1}\,\mathbf{v}^{\mathrm{f}}\right) \\ &= \mathrm{D}_{aPDL1}\nabla^2\mathrm{aPDL1} + \mathrm{P_{er}}\mathrm{S_V}\left(\mathrm{C_{iv}} - \mathrm{aPDL1}\right) + \mathrm{L_p}\mathrm{S_V}\left(\mathrm{P_V} - \mathrm{p_i}\right)\left(1 - \sigma_{\mathrm{f}}\right)\mathrm{C_{iv}}\end{aligned} \tag{20}$$

where C_{iv} is the vascular concentration of the antibody and is taken to be Civ = $\exp(-(t-t_0)/k_d)$ describing a bolus injection, with t_0 the time of drug injection and k_d the blood circulation decay of immunotherapy, σ_f is the reflection coefficient, P is the vascular permeability and L_p, S_v and p_v are the hydraulic conductivity, vascular density and vascular pressure, respectively

3. Technical improvements (specialization/modularity) of the DCoMEX-BIO codebase:

To abstract the definition and selection of the physics equations, a base class was introduced. This base class encapsulates all operations and properties specific to each advection-diffusion-reaction equation, providing a common structure that can be inherited and extended by specialized equation implementations.

```
public abstract class PhysicsModelBuilderBase : IPhysicsParaviewDataProvider
{
    public SolutionStrategies GetDefaultTimeSolutionPreferences()
    {
        return new SolutionStrategies(ParentAnalyzerType.GeneralizedA,
SolveForInitialConditions.DontSolve, TimeIntegrationOrderChoice.useFirstOrder);
    }
public List<(BC, ConvectionDiffusionDof[], double[][], double[])> DirichletBCs { get; set;
}
    public List<(BC, ConvectionDiffusionDof[], double[][], double[])> NeumannBCs { get;
set; }
    public List<(BC, ConvectionDiffusionDof[], double[][], double[])>
ConvectionDiffusionDirichletIC;

    public GlobalAlgebraicModel<CscMatrix> algebraicModel;
    public ComsolMeshReader Mesh { get; set; }

    public Dictionary<int, double[]> deformedStateNodeDisplacements { get; set; }

    public List<DataEntry> ComsolInitialConditionData;

    public SolutionStrategies solutionPreferences { get; set; }
```

In the code snippet, the aforementioned base class can be seen denoted “PhysicsModelBuilderBase” and it gathers properties such the ordering of the Degrees of freedom of the model the Boundary and initial conditions of the model, the mesh, user defined choices regarding the time integration algorithm and its properties and data regarding the current state of the model.

It also contains the implementation of the main required methods such as the construction of the analyzer objects, the update of its solution, the imposition of boundary conditions, the initialization of the solution structures and other methods as seen in the next code snippet from MGroup.DrugDeliveryModel application

```
public abstract class PhysicsModelBuilderBase : IPhysicsParaviewDataProvider
{
   . . .
   public (IParentAnalyzer analyzer, ISolver solver, IChildAnalyzer
   loadcontrolAnalyzer) GetAppropriateSolverAnalyzerAndLog
    (Model model, double pseudoTimeStep, double pseudoTotalTime, int currentStep)

    . . .

   private List<(INode node, IDofType dof)> GetTotalWatchDofs(Model model)

    . . .

   public List<double> RetrievePhysicsSolution(IChildAnalyzer childAnalyzer)

    . . .

   public void UpdateSharedQuantityDictionaryOriginal(IChildAnalyzer
childAnalyzer, Model     model, Dictionary<int, double> domainCox)

    . . .

   public virtual void AddInitialConditions(Model model)

    . . .

   public void AddBoundaryConditions(Model model)

    . . .
}
```

"Physics model builders" inherit from this base class and extend it to implement specialized methods such as the method "GetModel" that implements equation specific expressions and of course they hold the parameters that appear in each equation as well as the solution of the other fields that appear in the equation as well (concentration of other agents of the immune system or current interstitial fluid velocity etc.). In the following code snippet the appropriate model builder that implements the cancer cell concentration equation can be viewed and as it is illustrated it has as a property the concentration of the other agents of the immune system (elementwise Dictionaries). The implementation of the actual equation formula in the GetModel() method can also be seen.

```
public class TBuilder : PhysicsModelBuilderBase, IPhysicsBuilder {
    private double κ1 { get; }
    private double K2 { get; }
    private double D_T { get; }
    private double n_In { get; }
    private double n_ad { get; }
    private double κ_on { get; }
    private double KT { get; }
    private Dictionary<int, double> cox { get; }
    private Dictionary<int, double> aPDL1 { get; }
    private Dictionary<int, double> In { get; }
    private Dictionary<int, double> IAPC { get; }
    private Dictionary<int, double> TE { get; }
   public void AddBoundaryConditions(Model model)
public class TBuilder : PhysicsModelBuilderBase, IPhysicsBuilder {
. . .
public TBuilder(. . . . )
{
. . . .
this.productionFunctionParametricDefinitionTumor = new Func<double, double, double,
double, double, double, Func<double, double>>
((cox, aPDL1, In, IAPC, TE, divVsolid) => new Func<double, double>(T =>  k1*cox*T/(K2+cox)
- (n_In*In +n_In*IAPC+ n_ad*TE)*T - (k_on/KT)*aPDL1* (n_In * In + n_In * IAPC + n_ad * TE)
* T * T -divVsolid*T));
this.productionFunctionDerivativeParametricDefinitionTumor = new Func<double, double,
double, double, double, double, Func<double, double>>
((cox, aPDL1, In, IAPC, TE, divVsolid) => new Func<double, double>(T => k1 * cox / (K2 +
cox) - (n_In * In + n_In * IAPC + n_ad * TE) - 2*(k_on / KT) * aPDL1 * (n_In * In + n_In *
IAPC + n_ad * TE) * T - divVsolid ));
. . . .
      }
public Model GetModel()
{
var vs = SolidVelocity[elementConnectivity.Key];
                   convectionDomainCoefficients[elementConnectivity.Key] = new double[] {
             vs[0][0], vs[0][1], vs[0][2] };
independentProductionCoefficients[elementConnectivity.Key] = 0;
double divVsolid = div_vs is null ? 0 : div_vs[elementConnectivity.Key][0];
. . . . .
ProductionFuncWithoutConstantTerm[elementConnectivity.Key] = domainId == 0 ?
productionFunctionParametricDefinitionTumor(elementCox, elementAPDL1, elementIn, elementIAPC, elementTE,
divVsolid) : productionFunctionParametricDefinitionHost(elementCox, elementAPDL1, elementIn, elementIAPC,
elementTE, divVsolid);
ProductionFuncWithoutConstantTermDDerivative[elementConnectivity.Key] = domainId == 0 ?
productionFunctionDerivativeParametricDefinitionTumor(elementCox, elementAPDL1, elementIn, elementIAPC,
elementTE, divVsolid) : productionFunctionDerivativeParametricDefinitionHost(elementCox, elementAPDL1,
elementIn, elementIAPC, elementTE, divVsolid);
}
//Create Model
. . .
var model = modelProvider.CreateModelFromComsolFile(convectionDomainCoefficients,
diffusionCoefficient,
dependentProductionCoefficients, independentProductionCoefficients, capacity,
ProductionFuncWithoutConstantTerm, ProductionFuncWithoutConstantTermDDerivative);
return model;
}
}
```

Subsequently the coupled solution coordinator **`Coupled5eqNLProvider.cs`** was accordingly adjusted to account for a coupled model comprised of the PHG subproblem builder and an arbitrary number of advection-diffusion-reaction equation model builders. The refined class is named

```
public CoupledPhysicsModelFiniteStrain(PorousModelNonLinearProvider coupledModelProvider,
List<IPhysicsBuilder> PhysicsBuilders, ComsolMeshReader comsolReader, Dictionary<int,
double> domainCOx, Dictionary<int, double> T, Dictionary<int, double> lambda,
Dictionary<int, double[][]> pressureTensorDivergenceAtElementGaussPoints,
Dictionary<int, double[]> div_vs, Dictionary<int, double[]> FluidSpeed, Dictionary<int,
double[][]> solidVelocityAtElementGaussPoints,double kth_tumor, double timeStep, double
totalTime, int incrementsPerStep, double kth_host=0, Dictionary<int, double>
elementPressureAtSingleGPs = null, List<ITimeDependentBuilder> timeDependentBuilders =
null, bool includeSolidVelocityInFluidVelocity = false)
        {
            . . . .
            porousModelProvider = coupledModelProvider;
             physicsBuilders = PhysicsBuilders;
             . . . .

        }
```

4. Computational model – Validation results:

The validation procedure was comprised of a set of simplified models, closely following the development procedure steps gradually advancing towards the finalized model that is presented next. In the DrugDeliveryModel code repository this example can be found as: ImmunoModelingMultiInjection.cs (method RunGmsh1752WithImmunoWithStabil())

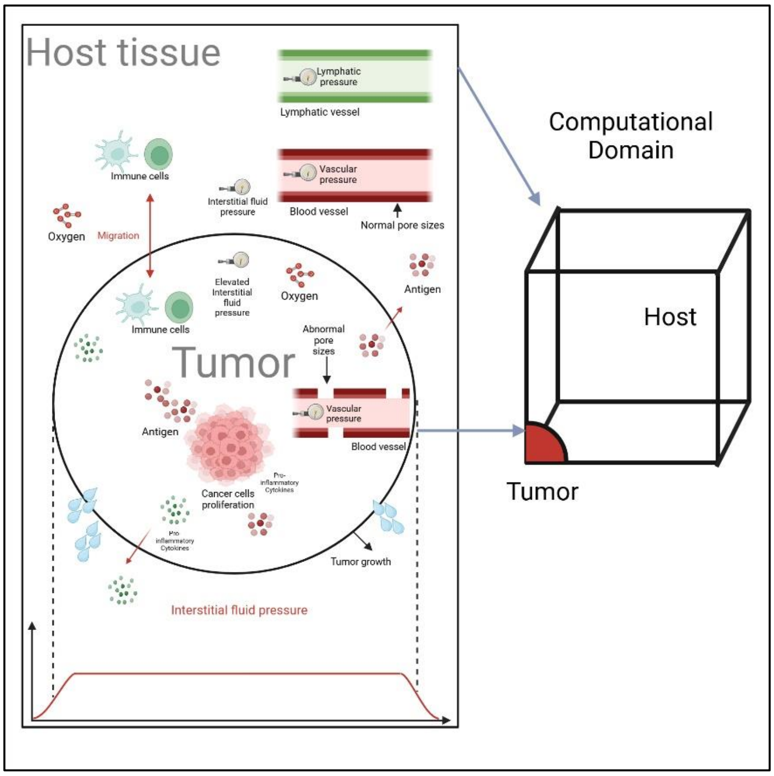

Figure 2 *Computational domain of the Immune cell and cancer cell interactions model*

The finalized benchmark model simulates the actual experimental applications, and is based on a two-region model consisting of a 3D rectangular host domain representing healthy tissue, with dimensions of 0.2 m × 0.2 m × 0.2 m. A spherical tumor region is nested at the center of the domain with a radius of 0.003 m. To reduce computational complexity, symmetry boundary conditions are applied, solving for one-eighth of the physical domain. The outer rectangular host tissue represents normal, healthy tissue, while the spherical region represents the tumor. Symmetry boundary conditions are applied on the symmetry planes, where the normal displacements for the hyperelastic model are fixed, and only slip degrees of freedom are permitted in the tangential directions. For the convection-diffusion-reaction equations that predict the concentration of the immune system agents, zero flux is imposed on the symmetry planes, ensuring no mass transfer across these boundaries. Dirichlet boundary conditions are applied to the outer planes of the cube, fixing the concentrations on these faces.

A time-dependent solver is employed, with very fine time-stepping in the begging and gradually increasing, as described from the following ranges and corresponding time steps sizes: {(0, 0.2, 40) seconds, (40, 40, 4000) seconds, and range(4000, 2000, 1209600) seconds}, effectively solving the problem over a total of 14 days, equivalent to 1,209,600 seconds. The initialization of the simulation corresponds to a day of the experiment where the tumor is large enough to be model by continuum mechanics theory. The computational mesh consists of 1,730 vertices, resolving a total of 34,774 degrees of freedom, with an additional 11,459 internal degrees of freedom.

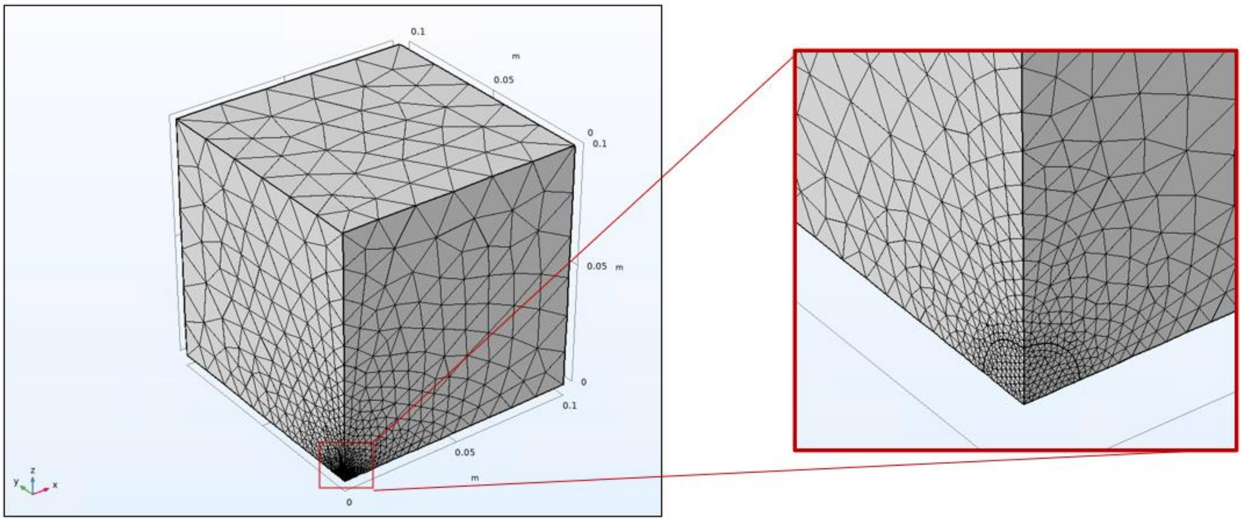

Figure 3: Benchmark Application discretized domain

Anti-PD-L1 injections are modeled as transient events, occurring on days 7, 10, and 13 (or 7,9 and 11 depending on the studied experiment). The initial concentrations of cancer cells and immune system agents are provided in Table 1, while the initial stretch value for the growth equation ODE is set to 1, indicating no initial deformation of the tissue.

The solver is configured using a Generalized Alpha Solver, with a second-order scheme applied to the porous model and a first-order scheme for both the growth ODE and the advection-diffusion-reaction equations. The solver uses a Newton-Raphson iterative solution procedure to handle nonlinearities that arise in the equations. The anti-PD-L1 injection profile is defined by a step function, with a continuous derivative up to the second order and an exponential decay, to simulate drug release.

The model is validated by comparing the results with a commercial software (COMSOL) widely used for such models. Specifically, validation is performed by comparing the time-history solution at a set of strategically placed logged nodes, starting from the tumor region and extending outward into the healthy tissue, allowing for a comprehensive assessment of tumor progression and immune response. A detailed list of the model parameters can be found in Table 1 of the present report.

our nodes are considered foreach studied field: T, p , λ_g, T_E, In, IAPC, cox, c, aPDL1, APC, Ag, u_z while X-axis values' units are "s" seconds

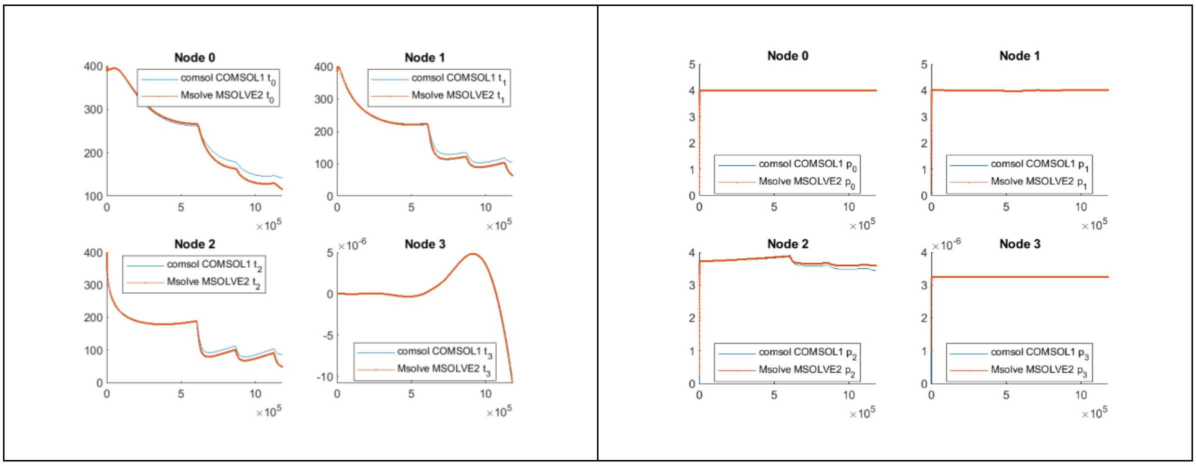

***Figure 4a,b** :Comparison of MSolve and COMSOL soution at four selected nodes: T and p equations*

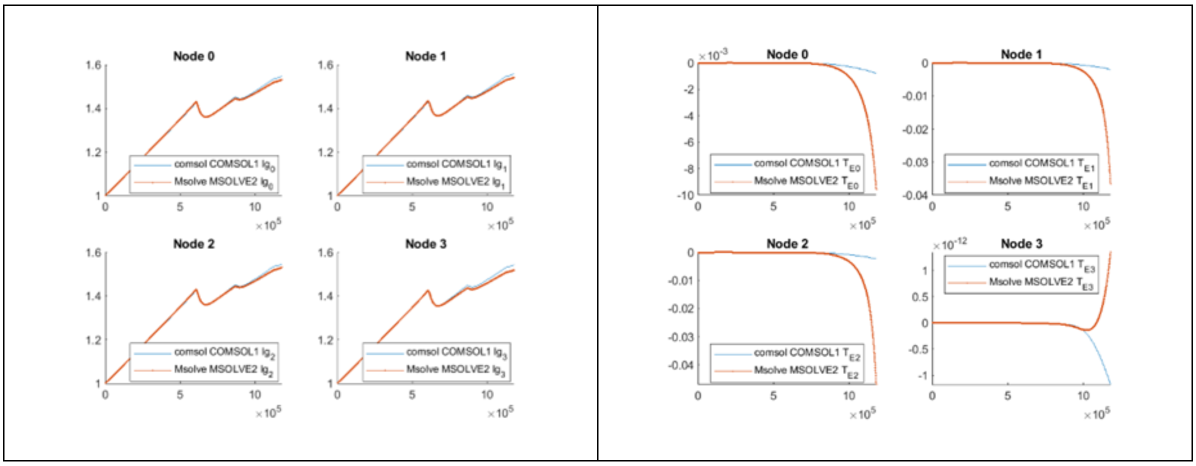

***Figure 5a,b** :Comparison of MSolve and COMSOL soution at four selected nodes: λ_g and T_E equations*

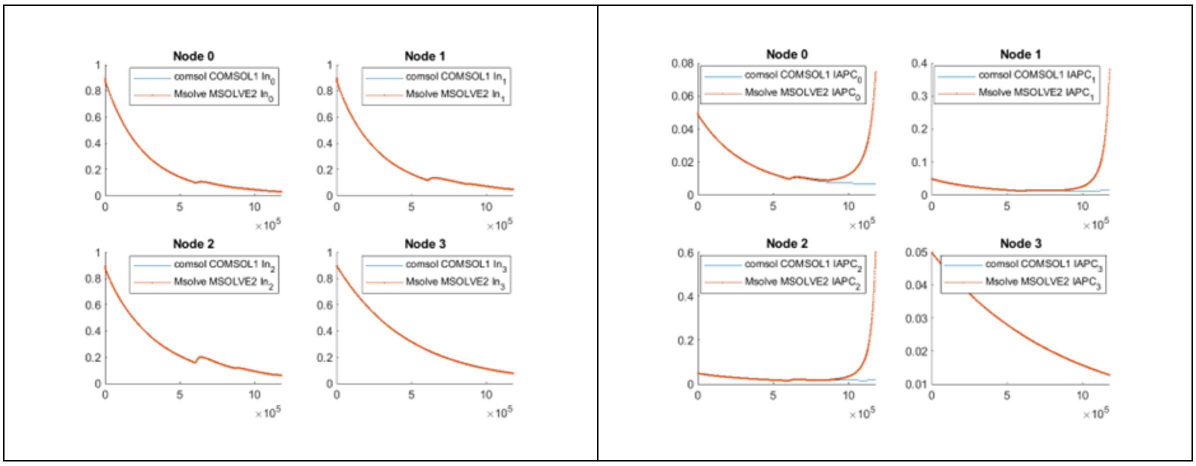

***Figure 6a,b** :Comparison of MSolve and COMSOL soution at four selected nodes: I_n and IAPC equations*

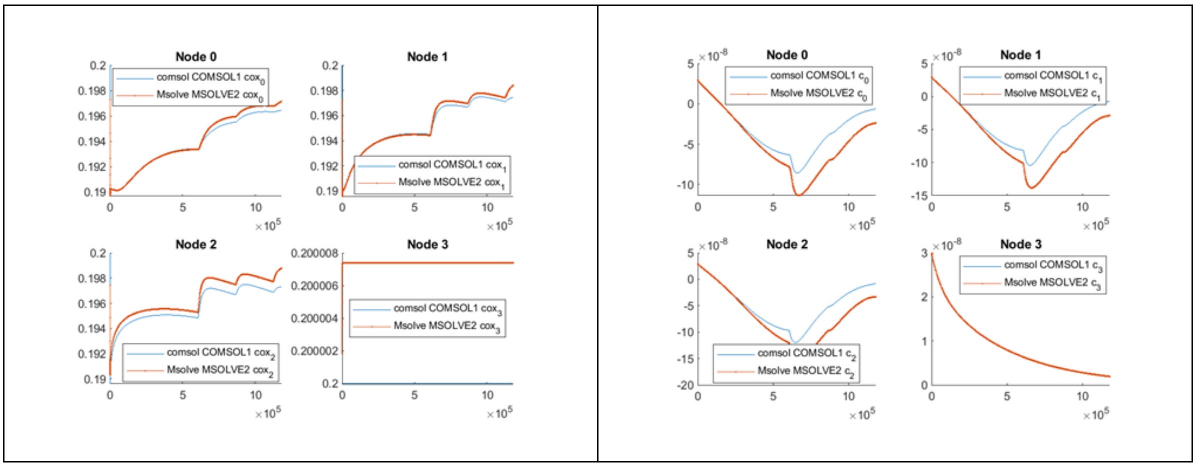

***Figure 7a,b** :Comparison of MSolve and COMSOL soution at four selected nodes: I_n and IAPC equations*

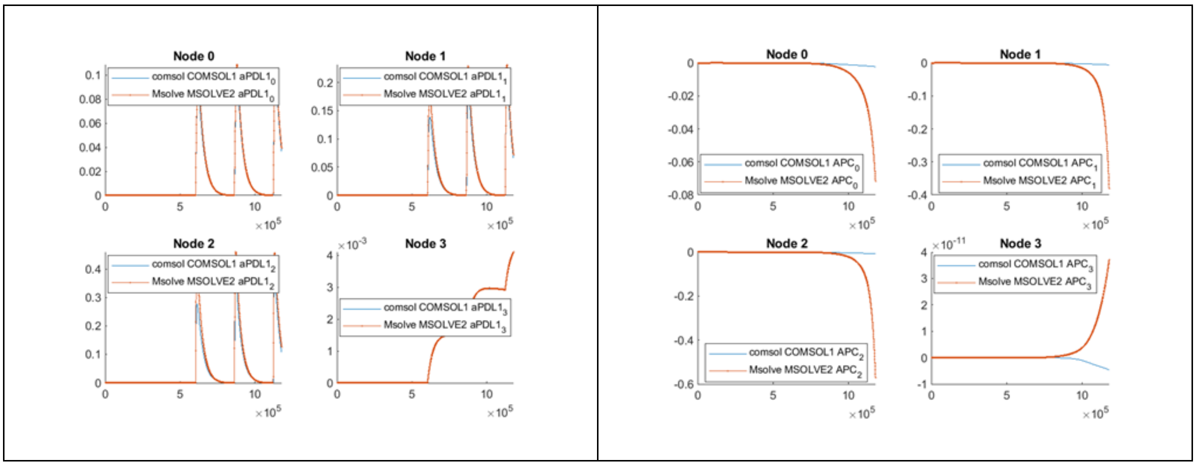

***Figure 8a,b** :Comparison of MSolve and COMSOL soution at four selected nodes: aPDL1 and APC equations*

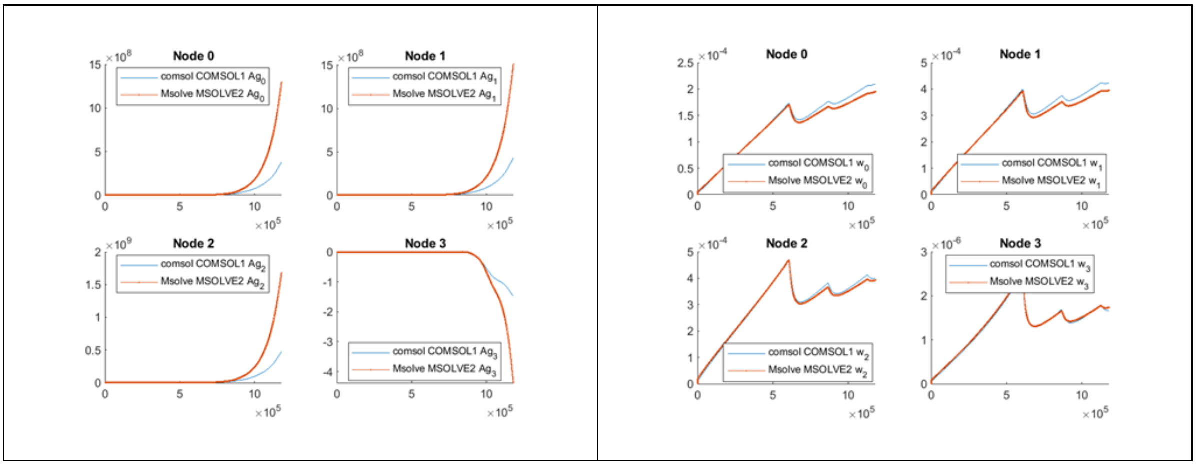

***Figure 9a,b** :Comparison of MSolve and COMSOL soution at four selected nodes: I_n and IAPC equations*

5. Experimental data

The original experiments that were replicated computationally in this project, are described in the following paper:"**Voutouri C, Mpekris F, Panagi M, Krolak C, Michael C, Martin JD, Averkiou MA, Stylianopoulos T.** Ultrasound stiffness and perfusion markers correlate with tumor volume responses to immunotherapy. *Acta Biomaterialia*. 2023;167:121-134. doi:10.1016/j.actbio.2023.06.007"

The simulated time point in MSolve denoted as start of the simulation refers to a later point in the experimental procedure when the tumor is large enough to be modeled by continuum mechanics theory. This offset value is given as day 10 of the experiment for 4T1 and E0071 tumors, day 4 for MCA205 tumors and day 7 for B16F10.

6. Surrogate model development

Problem Representation

The task is to create a surrogate model that approximates the solution manifold of the stochastic tumor microenvironment model efficiently while bypassing the high computational cost associated with traditional solvers utilized in the high-fidelity FEM method.

6.1 Data Generation for Surrogate Modeling

In accordance with the classical Monte Carlo simulation workflow, a high-fidelity finite element method (FEM) model was used to generate the training datasets. For each simulation, input parameters were sampled from their respective probability distributions, ensuring adequate representation of the stochastic nature of the problem.

It is important to note that the subsequent dataset partitions result in a smaller proportion of data allocated to training compared to testing. This approach is deliberate, as the primary objective is to minimize the size of the training set. The rationale behind this strategy is to prioritize the use of the trained neural network for evaluating the test data, thereby achieving a significant speedup in the overall stochastic or optimization analysis.

In this regard, a 4-fold partition was initially considered in the form of training, validation, test and analysis/production datasets, but since the operational difference between the test and analysis datasets would be miniscule, we ultimately resolved to adhere to the classical train, validation, test partition scheme.

Dataset 1: 5 Equations

The chosen, uniformly distributed, stochastic parameters were the following:

- Shear Modulus: $\mu \sim U[5,50]KPa$
- Vascular Density: $Sv \sim U[5 \cdot 10^3, 2 \cdot 10^4]\, 1/m$
- Hydraulic Conductivity: $K_\theta \sim U[7.52 \cdot 10^{-12}, 7.52 \cdot 10^{-10}]\, m^2/(KPa \cdot sec)$

This process produced a dataset consisting of 2,000 samples. The dataset was then partitioned into training and test sets, with 20% of the samples allocated to training, 10% for validation and the remaining 70% reserved for testing. The DCoMEX-BIO prototype model was used for this dataset that accounts for 5 equations (displacement, pore pressure, oxygen concentration, growth equation, cancer cells concentration equation) and is described in report D5.2.

Dataset 2: 12 Equations with 3-stage Immunotherapy

The dataset was produced by solving the full model of 12 equations with immunotherapy sessions at 3 discrete time points:

1. First session at 7 days
2. Second session at 10 days
3. Third session at 13 days

The chosen, uniformly distributed, stochastic parameters were the following:

1. Shear Modulus: $\mu \sim U[5,50]KPa$
2. Vascular Density: $Sv \sim U[5 \cdot 10^3, 2 \cdot 10^4]\, 1/m$
3. Hydraulic Conductivity: $K_\theta \sim U[7.52 \cdot 10^{-12}, 7.52 \cdot 10^{-10}]\, m^2/(KPa \cdot sec)$
4. $pv \sim U[0.6667,6.667]KPa$
5. $k_1 \sim U[\, 1.574 \cdot 10^{-6}, 3.988 \cdot 10^{-6}]\, 1/sec$
6. $Lp \sim U[\, 1.56 \cdot 10^{-8}, 8 \cdot 10^{-7}]\, \frac{m}{KPa \cdot sec}$
7. $sf \sim U\left[0.136\square^{\square}, 0.146^{\square}\right]$
8. $Per \sim U[4.16 \cdot 10^{-9}, 6.225 \cdot 10^{-8}]\, m/s$
9. $K_T \sim U[1.273 \cdot 10^{-6}, 1.273 \cdot 10^{-4}]\, \frac{kg}{m^3 \cdot sec}$
10. $k_{on} \sim U[1.157 \cdot 10^{-8}, 1.157 \cdot 10^{-2}]\, \frac{m^3}{kg \cdot sec}$
11. $kd \sim U[2.998 \cdot 10^4, 9.297 \cdot 10^4] sec$

Dataset 3: 12 Equations with Immunotherapy

The dataset was produced by solving a model of 12 equations with immunotherapy sessions at 3 discrete time points:

1. First session at 7 days
2. Second session at 9 days
3. Third session at 11 days

The chosen, uniformly distributed, stochastic parameters were the following:

1. Shear Modulus: $\mu \sim U[5,50]KPa$
2. Vascular Density: $Sv \sim U[5 \cdot 10^3, 2 \cdot 10^4]\, 1/m$
3. Hydraulic Conductivity: $K_\theta \sim U[7.52 \cdot 10^{-12}, 7.52 \cdot 10^{-10}]\, m^2/(KPa \cdot sec)$
4. $pv \sim U[0.6667,6.667]KPa$
5. $k_1 \sim U[\, 1.574 \cdot 10^{-6}, 3.988 \cdot 10^{-6}]\, 1/sec$
6. $Lp \sim U[\, 1.56 \cdot 10^{-8}, 8 \cdot 10^{-7}]\, \frac{m}{KPa \cdot sec}$
7. $sf \sim U\left[0.136\square^{\square}, 0.146^{\square}\right]$
8. $Per \sim U[4.16 \cdot 10^{-9}, 6.225 \cdot 10^{-8}]\, m/s$
9. $K_T \sim U[1.273 \cdot 10^{-6}, 1.273 \cdot 10^{-4}]\, \frac{kg}{m^3 \cdot sec}$
10. $k_{on} \sim U[1.157 \cdot 10^{-8}, 1.157 \cdot 10^{-2}]\, \frac{m^3}{kg \cdot sec}$
11. $kd \sim U[2.998 \cdot 10^4, 9.297 \cdot 10^4] sec$

Dataset 3: 12 Equations without Immunotherapy

The following parameters remained constant:

1. $sf = 0.136$
2. $Per = 4.16 \cdot 10^{-9}\, m/s$
3. $K_T = \; 1.243 \cdot \; 10^{-6} \frac{kg}{m^3 \cdot sec}$
4. $k_{on} = \; 1.157 \cdot \; 10^{-8} \frac{m^3}{kg \cdot sec}$
5. $kd = 2.998 \cdot 10^4 sec$

The chosen, uniformly distributed, stochastic parameters were the following:

1. Shear Modulus: $\mu \sim U[5,50]KPa$
2. Vascular Density: $Sv \sim U[5 \cdot 10^3, 2 \cdot 10^4]\, 1/m$
3. Hydraulic Conductivity: $K_\theta \sim U[7.52 \cdot 10^{-12}, 7.52 \cdot 10^{-10}]\, m^2/(KPa \cdot sec)$
4. $pv \sim U[0.6667,6.667]KPa$
5. $k_1 \sim U[\, 1.574 \cdot 10^{-6}, 3.988 \cdot 10^{-6}]\, 1/sec$
6. $Lp \sim U[\, 1.56 \cdot 10^{-8}, 8 \cdot 10^{-7}]\, \frac{m}{KPa \cdot sec}$

6.2 Discrete Time Point Prediction Surrogate

The goal was to construct and train a surrogate model that accurately predicts the value of the tumor region in discrete points in time given the value of stochastic parameters (3 parameters for the 5 equation model and 11 or 6 parameters for the advance model that accounts for the immune system reaction).

Surrogate Model: FFNN Architecture

To efficiently emulate the behavior of the system, a Feedforward Neural Network (FFNN) was chosen as the surrogate model. After thorough experimentation an optimal architecture was produced with respect to the accuracy of the network.

The FFNN consists of multiple stacked blocks, each composed of:

1. *Batch Normalization*: Normalizes the input to a layer to stabilize training and reduce sensitivity to hyperparameter tuning.
2. *Linear Layer*: Performs linear transformations based on learned weights and biases.
3. *Leaky ReLU Activation*: Introduces non-linearity and mitigates possible issues related to the vanishing gradient problem.
4. *Final Layer*: The final linear layer transforms the internal representation of the data into the predicted output.

Training Phase

The surrogate model was trained on the input-output data pairs using the following parameters:

- *Optimizer*: the Adam Optimizer, which adapts the learning rate for each parameter based on estimates of first and second moments of the gradients. This results in faster convergence.
- *Loss Function*: the Mean Squared Error (MSE), which measures the squared differences between the predicted and actual simulation results. Minimizing this loss ensures that the FFNN accurately predicts the system response.
- *Number of Epochs*: the network is trained for 8,000 epochs
- *Batch size*: the input data were given as batches 128 ensuring a comprehensive search for optimal weights and biases across many iterations.

Validation and Performance Evaluation

After training, the performance of the surrogate models was evaluated with the Relative Root Mean Square Error (RRMSE) on the test set.

1. Final Volume prediction model at **Dataset 1**: RRMSE = 2%

Given the empirical engineering error threshold of 5%, the trained surrogate models produce predictions that are highly accurate relative to the original, exact simulations at a minuscule fraction of the high-fidelity model's execution time.

6.3 Full Timeseries Response Surrogate: Hybrid CAE-FFNN Architecture

The surrogate model consists of a composite architecture that leverages the representational and feature-extraction power of *Convolutional Auto-Encoders* (CAEs) combined with classical feedforward neural networks (FFNNs).

- *Encoder Network*: The component is adept at capturing localized patterns and spatial hierarchies within the input data. In the context of parametric ODEs, it was utilized to extract latent spatio-temporal features from discretized representations of the system's solution space and encode them into a low-dimensional latent space.
- *FFNN*: The FFNN is responsible for learning the global parameterized mapping from stochastic input parameters to the CAE's latent space.

- *Decoder Network:* This component takes the low-dimensional latent space representation of the system's solution space, as encoded by the Encoder part, and refines them through a series of inverse pooling and convolutional networks. Finally, it projects the latent features into the desired solution space. In combination with the FFNN, they define a mapping from input parametric space to the solution space.

To improve computational efficiency, the surrogate model performs a dimensionality reduction via an information-preserving latent space. The CAE architecture aims to capture the essential dynamics of the ODE system in this reduced space.

Architecture

- The *FFNN* consists of 2 hidden layers of 500 neurons with the GELU activation function.
- The *Auto-Encoder* consists of one dimensional convolution, max pooling and layer-normalization layers, followed by the GELU activation function, applied in succession.
  The latent space had a dimension of 6.

Training Phase

The training phase is divided into two parts:

1. CAE training: the Autoencoder is trained by learning to match its output with the input solution timeseries creating the latent space representation in the process.
2. FFNN training: the Feed-Forward network is trained to map elements of the stochastic parameters space to the latent CAE space.

Inference Phase

During inference, the surrogate model bypasses the high-cost FEM solver entirely, thus significantly reducing the computational burden. Firstly, the FFNN takes as input the stochastic parameters' vector and maps it to a point in the latent space. Then, the Decoder network upsamples this projection to produce a timeseries approximaiton of the solution response.

Performance Metrics

The evaluation of the surrogate model for the 5 equations was performed on a timeseries dataset consisting of 2000 samples of solution vectors over 850 timesteps. Each physical field of the problem was treated separately with a different CAE model.

The surrogate's efficiency and accuracy were evaluated based on the *relative root mean square error* (RRMSE), with values of approximately 3% on the test set, while achieving a speedup S=4.94 on the total execution time of the stochastic analysis compared to the non-surrogate approach.

Dataset Creation

The analyses involved in the creation of the surrogates were executed in MELUXINA Supercomputer infrastructure. More details about the runs are given in the performance evaluation report for DCOMEX-BIO in D.8.1.

6.4 Conditional Neural Field Surrogate for Dataset 2

For Dataset 2, we trained a Neural Field to predict the timor volume conditioned on both the 11 input parameters as well as the current time. The architecture chosen was inspired by the DeepONet architecture. In more detail the parameters were first processed using five fully connected layers with widths 11,30,50,50 and 50 whereas the current time is input to four layers of width 1,10,50,50. Subsequently the final inputs of both these networks are combined using a scalar product. We note that variations in the architecture choice were also explored and did not lead to significant changes in the final results.

The chosen architecture was trained using the Adam Optimizer using 70 percent of the data set while the remaining data points were used for either validation or testing.

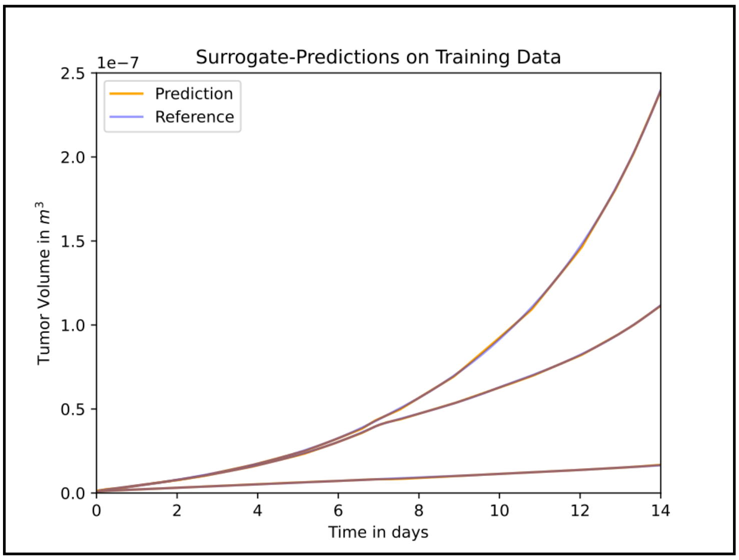

***Figure 10** Conparison of solution of computational model and surrogate prediction of a sample point of Training Data*

On the training data set, we achieved a RMSE of 0.6 percent averaged over all time steps.

On the test data set, we achieved a RMSE of 4.7 percent averaged over all time step. Especially the drastic change in the systems dynamics after the treatment appeared to be challenging for the surrogate. As the error is still below 5 percent and thus the threshold we selected, the results were still acceptable especially given that this surrogate is able to offer significant speed up compared to the original model

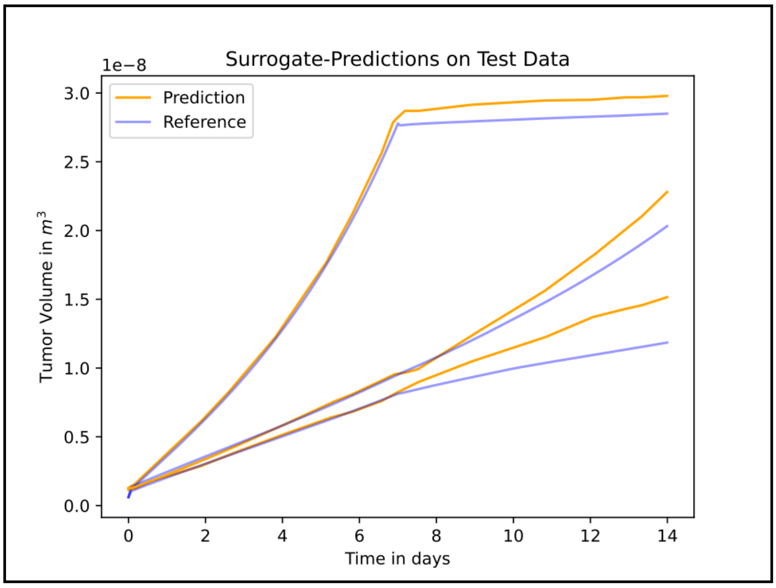

***Figure** 11 Comparison of solution of computational model and surrogate prediction of a sample point of Test Data*

7. Bayesian inference

7.1. Proof of Concept on 5-equation model with three parameters

After we have replaced the 5-equation-model by a surrogate model as described in the previous section to ensure a fast evaluation, we inferred the parameters of the model given some observations using a Bayesian inverse analysis.

To do so, we assumed uniform priors on the three model parameters and solved a Bayesian inverse problem. We sampled from the posterior using the Korali implementation of tMCMC. Whereas this leads to the necessity of one surrogate model evaluation (in the likelihood computation) for each sample, we can manage to keep the computational cost on a reasonable level due to the surrogate model.

Illustrative results for one test case can be found below. The single observed data point is marked with a red x and the posterior encapsulates this data points. We observe that the influence of the parameters k_1 and Sv is small compared to µ which is indicated by the shape of the posterior.

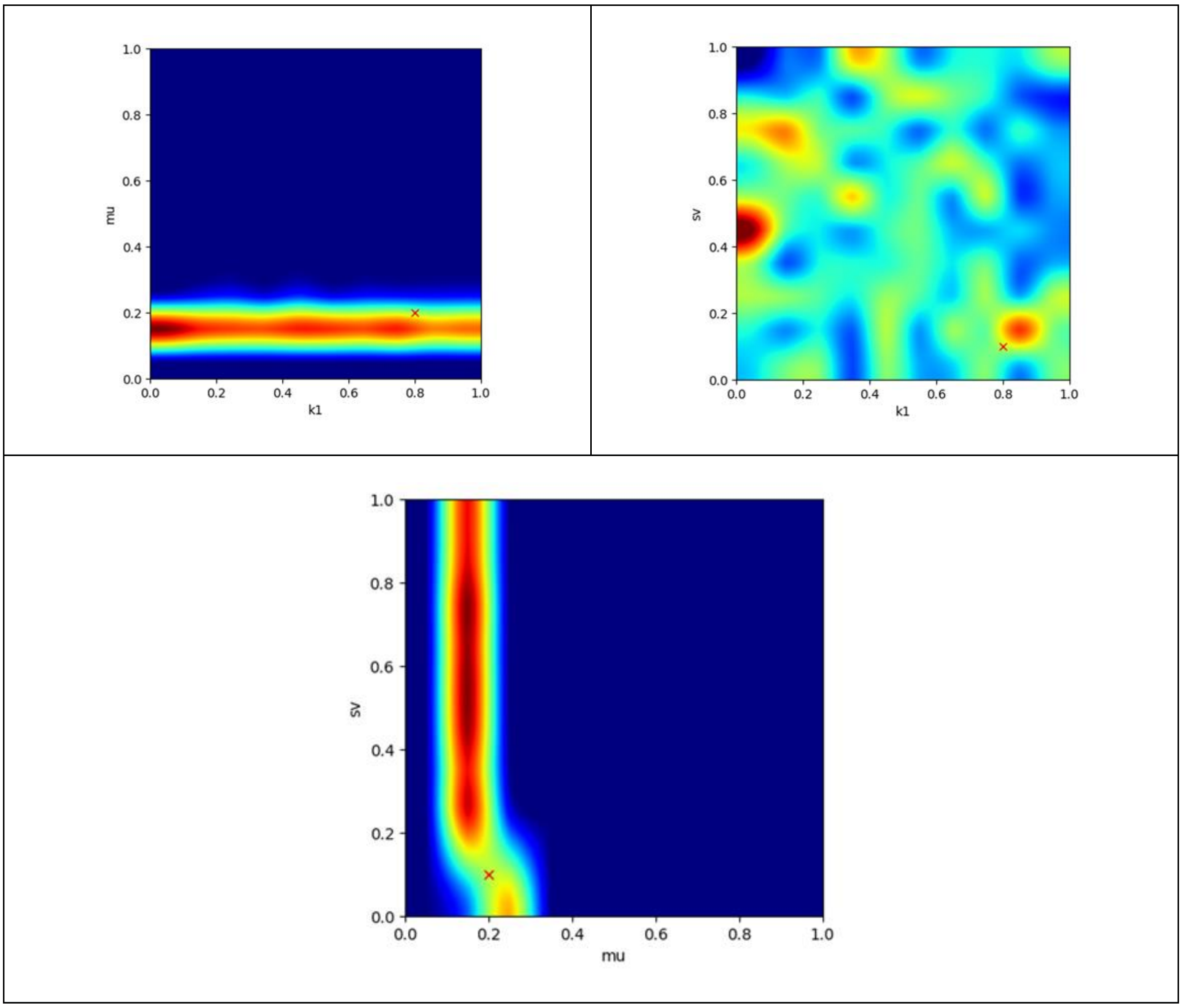

Figure 12 a,b *and* ***c*** *: influence of the parameters mu k1 and sv*

7.2. Parameter Inference for the experimental data using a Surrogate Model

Based on the conditional Neural Field Surrogate Model for Dataset 2 of the 12 equation model with 11 parameters, we solved the Bayesian Inverse Problem of finding the posterior for the parameters given experimental data. We generated samples from this posterior using TMCMC. Despite only scarce and noisy data points we are able to learn a distribution of parameters that is able to fit the data of interest.

The Figure shows the experimental data (red crosses), the posterior mean of the learned model (blue) as well as the 20 and 80 quantile of the predictive uncertainty of this model. This indicates that the model can be successfully calibrated using experimental data.

We are moreover reporting some illustrative marginal distributions for the obtained parameters.

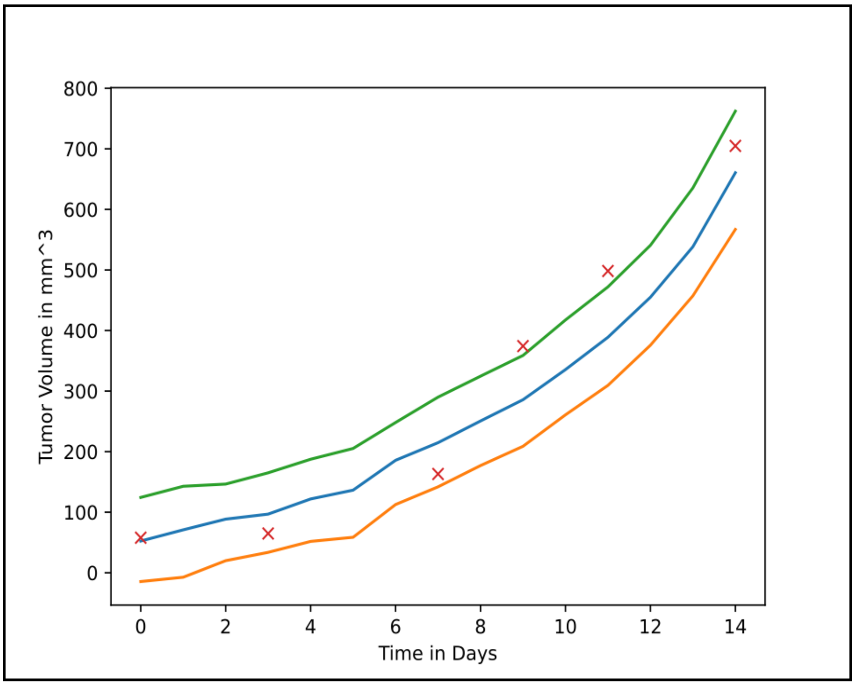

Figure 13 *Mean value response, 80th and 20th quantiles response of the learned posterior distribution*

Here, sigma is the parameter for the noise of the experimental data, whereas the other two figures show two of the eleven parameters. All values are normalised between 0 and 1.

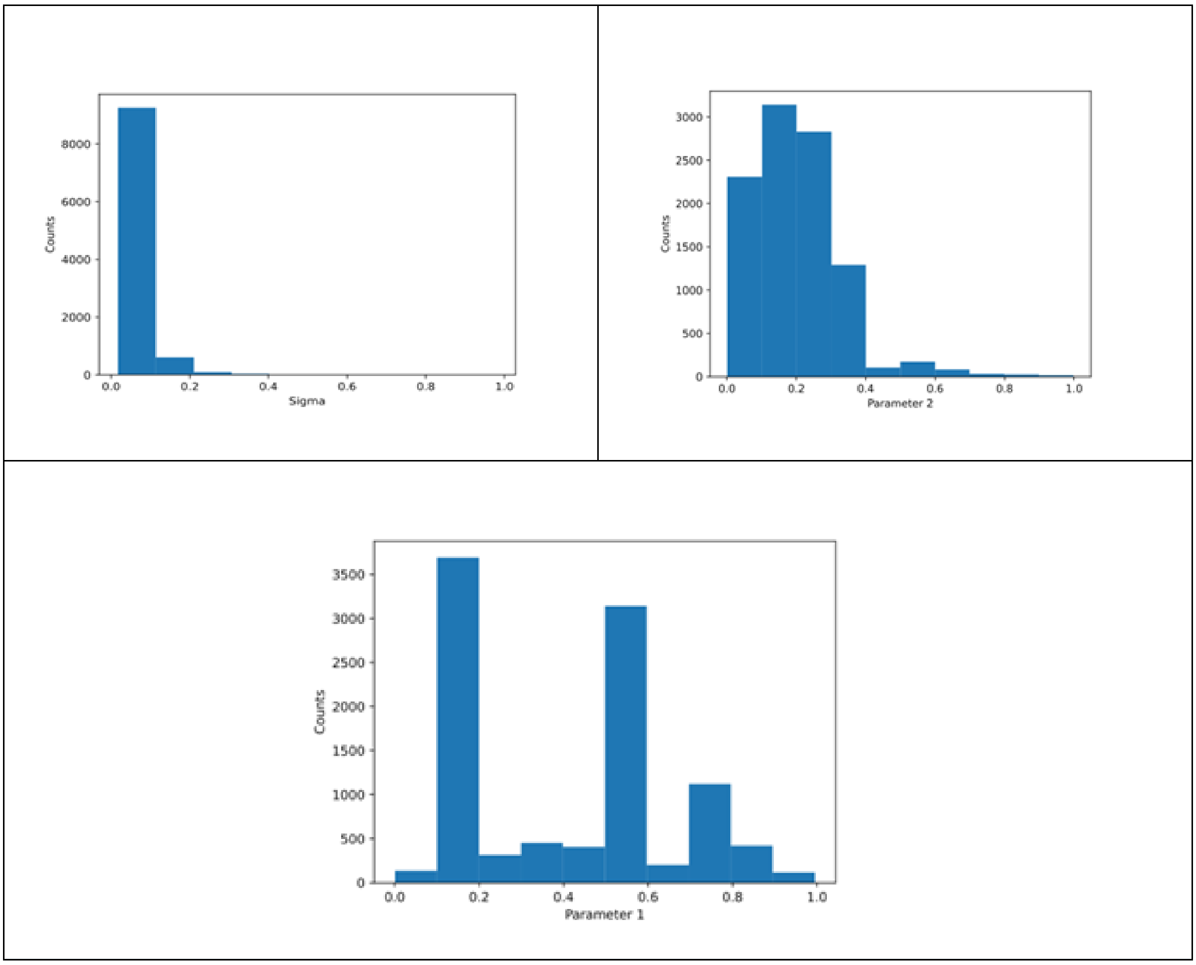

Figure 14a: *posterior distribution of the noise parameter, sigma,* ***14b:*** *posterior distribution of parameter Sv,* ***14c:*** *posterior distribution parameter μ,*

8. Optimization

Deterministic case:

For a set of determined constitutive parameters of the model, the response of the system relies only of the discrete time points of the immunotherapy injection. And the volume of the tumor region at the end of the computational experiment (14 days) can be expressed as

$$\mathrm{V}_{14} = F(T_1, T_2, T_3)$$

In order to obtain an optimal time sequence of the immunotherapy injection we aim to minimize the function $F(T_1, T_2, T_3)$ subject to the following constraints:

- $T_1 \ \in \{\,7, 8, 9, 10, 11\}$ that represents the first injection
- $T_2 \ \in \{\,8, 9, 10, 12\}$ that represents the second injection with the constraint $T_2 > T_1$
- $T_3 \ \in \{9, 10, 13\}$ that represents the third injection with the constraint $T_3 > T_2$

Thus the optimization problem is:

$$\textit{Minimize } F(T_1, T_2, T_3)$$

Subject to:

$$T_1 \ \in \{\,7, 8, 9, 10, 11\} \qquad T_2 \ \in \{T_1 + 1\,, \dots\dots, 12\} \qquad T_3 \ \in \{\,T_1 + 1, \dots\dots, 13\}$$

It is a problem with three design parameters with discrete values. Appropriate optimization algorithms for that case include the Simulated Annealing (SA) , Genetic algorithms (GA) and others (Particle Swarm Oprimization (PSO etc.). However, the specified design space dimensions are very limited and results in only 20 possible combinations of the design parameters values. For that reason we will execute an Exhaustive Search for the given set of parameters and no use of a NN is necessary. Figure 10 depicts the results of the time evolution of tumor volume value for each time point combination.

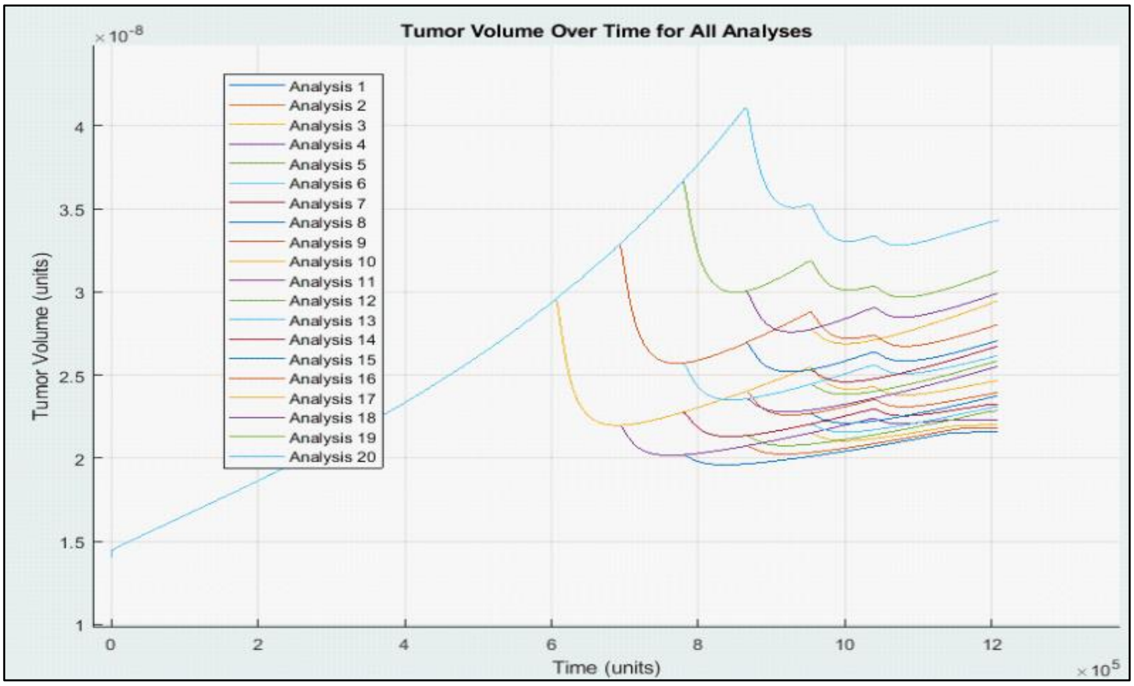

Figure 15 *Time evolution of tumor region volume for different immunotherapy time sequences*

The minimum volume value is obtained as 172.5 mm^3 for the time sequence of injections {7, 8, 9 days}.

Supplementary Tables

Supplementary Table S1. Parameters and their values used in the model

Parameter	Description	Value	Reference
μ	shear modulus	5.00 kPa for normal tissue; 22.44 kPa for tumor	[12–14]
K	Bulk modulus	6.667 kPa for normal tissue 216.7 kPa for tumor tissue	
k_{th}	hydraulic conductivity	6.5×10^{-11} m^2/Pa·day for normal tissue: 6.5×10^{-9} m^2/Pa·day for tumor tissue	[14]
S_v	Vascular density	7000 m^{-1} for normal tissue 7000 m^{-1} for tumor tissue	[15]
C_{iox}	initial oxygen concentration	0.2 mol·m^{-3}	[16]
D_{ox}	oxygen diffusion coefficient	1.55×10^{-4} m^2·day^{-1} for tumor tissue 1.55×10^{-4} m^2·day^{-1} for normal tissue	[2]
P_{erox}	Oxygen permeability across tumor vessel wall	3.55×10^{-4} m/s for normal and 3.55×10^{-4} m/s for tumor tissue	[17]
A_{ox}	oxygen uptake	2,200 mol·m^{-3}·day^{-1} for normal and	[2,16]

		2,200 mol·m^{-3}·day^{-1} for tumor tissue	
k_{ox}	oxygen uptake	0.00464 mol·m^{-3}	[2,16]
k_1	growth rate parameter	0.3446 1/day	[7]
k_2	growth rate parameter	0.0083 mol·m^{-3}	[16]
p_v	Vascular pressure	30 mmHg [4kPa]	[9]
$L_{pl}S_{vl}$	Permeability of lymphatic vessels	3.75 x10^{-4} [1/(Pa-s)] for normal tissue 0 for tumor tissue	[15]
p_l	lymphatic vessels pressure	0	[15]
vis	Water viscosity at 310K	$7 \cdot 10^{-4}$ Pa · s	[18]
k_{In}	Production of proinflammatory cytokines by innate immune cells	$3 \cdot 10^{-8}$ day^{-1}	[19]
k_{TE}	Production of proinflammatory cytokines by effector CD8+ Tcells	$3 \cdot 10^{-8}$ day^{-1}	[19]
k_{APC}	Production of proinflammatory cytokines by antigen presenting cells	$3 \cdot 10^{-8}$ day^{-1}	[19]
δ_{APC}	Degradation of antigen presenting cells	0.1 day^{-1}	[19]
δ_{IAPC}	Degradation of immature antigen presenting cells	0.1 day^{-1}	[19]

δ_{TE}	Initial degradation of effector CD8+ Tcells	$0.18\ \mathrm{day}^{-1}$	[19]
δ_{In}	Initial degradation of Innate immune cells	$0.18\ \mathrm{day}^{-1}$	[19]
δ_{c}	Degradation of cytokines produced by immune cells	$1.38\ \mathrm{day}^{-1}$	[19]
χ_{APC}	Production of APCs	$0.5\ \mathrm{cm}^{3} \cdot \mathrm{g}^{-1}$	[7]
m_{APCM}	Source of effector CD4+ and source of effector CD8+ (Momin et. al.)	$2.36 \cdot 10^{-6}\ \mathrm{s}^{-1}$	[7]
λ_{In}	Production of NK	$0.025\ \mathrm{g} \cdot \mathrm{cm}^{-3} \cdot \mathrm{day}^{-1}$	[7]
λ_{IAPC}	Production of IAPC	$0.025\ \mathrm{g} \cdot \mathrm{cm}^{-3} \cdot \mathrm{day}^{-1}$	[7]
K_{cIn}	Half saturation concentration Innate cells	$9.1346 \cdot 10^{-4}\ \mathrm{g} \cdot \mathrm{cm}^{-3}$	[7]
K_{cAPC}	Half saturation antigen presenting cells	$9.1346 \cdot 10^{-4}\ \mathrm{g} \cdot \mathrm{cm}^{-3}$	[7]
n_{In}	Killing rate constants of tumor cells by innate immune cells	$54.1857\ \mathrm{cm}^{3} \cdot \mathrm{g}^{-1} \cdot \mathrm{day}^{-1}$	[7]
n_{ad}	killing rate constants of tumor cells by adaptive immune cells	$108.3713\ \mathrm{cm}^{3} \cdot \mathrm{g}^{-1} \cdot \mathrm{day}^{-1}$	[7]

$\mathrm{n}_{\mathrm{A_g}}$	Antigen uptake rate	$108.3713\ \mathrm{cm}^3 \cdot \mathrm{g}^{-1} \cdot \mathrm{day}^{-1}$	[7]
$\mathrm{D}_{\mathrm{IAPC}}$	Diffusion coefficient IAPC	$5.11 \cdot 10^{-13}\ \mathrm{m}^2 \cdot \mathrm{s}^{-1}$	[20]
$\mathrm{D}_{\mathrm{APC}}$	Diffusion coefficient APC	$5.11 \cdot 10^{-13}\ \mathrm{m}^2 \cdot \mathrm{s}^{-1}$	[20]
$\mathrm{D}_{\mathrm{T^E}}$	Diffusion coefficient CD8	$5.11 \cdot 10^{-13}\ \mathrm{m}^2 \cdot \mathrm{s}^{-1}$	[20]
D_{In}	Diffusion coefficient Innate immune cells	$5.11 \cdot 10^{-13}\ \mathrm{m}^2 \cdot \mathrm{s}^{-1}$	[20]
$\mathrm{D}_{\overline{\mathrm{T}}}$	Diffusion cancer cells	$5.11 \cdot 10^{-13}\ \mathrm{m}^2 \cdot \mathrm{s}^{-1}$	[20]
D_{c}	Diffusion coefficient pro-inflamatory cytokines by immune cells	$6.0472 \cdot 10^{-2}\ \mathrm{cm}^2 \cdot \mathrm{day}^{-1}$	[21]
$\mathrm{D}_{\mathrm{A_g}}$	Diffusion coefficient Antigen	$6.0472 \cdot 10^{-2}\ \mathrm{cm}^2 \cdot \mathrm{day}^{-1}$	[21]
Φ	Volume fraction of tumor accessible to drug	0.3	[22,23]
$\mathrm{c}_{\mathrm{cin}}$	Initial concentration of pro inflammatory cytokines	$3 \cdot 10^{-11}\ \mathrm{g} \cdot \mathrm{cm}^{-3}$	[19]
$\mathrm{In}_{\mathrm{In}}$	Initial concentration of Innate immune cells	$9 \cdot 10^{-4}\ \mathrm{g} \cdot \mathrm{cm}^{-3}$	[19]

$\overline{T_0}$	Initial concentration of tumor cells	$0.4\ \mathrm{g} \cdot \mathrm{cm}^{-3}$	[19]
k_{on}	Binding rate constant	$6.87 \cdot 10^{3}\ \mathrm{cm}^{3} \cdot \mathrm{g}^{-1} \cdot \mathrm{day}^{-1}$	[8]
D_{aPDL1}	Diffusion coefficient anti-PDL1	$7.85 \cdot 10^{-2}\ \mathrm{cm}^{2} \cdot \mathrm{day}^{-1}$	[24]
k_d	Blood circulation decay	0.417 day	[25]
t_0	Time of drug injection	14 day	This study
σ^f	Reflection coefficient	0.13427 for tumor 0.80829 for host tissue	Equation below
P_{er}	Vascular permeability	$0.13427 \cdot 10^{-11}\ \mathrm{m} \cdot \mathrm{s}^{-1}$ for tumor $2.69 \cdot 10^{-8}\ \mathrm{m} \cdot \mathrm{s}^{-1}$ for host tissue	Equation below
L_p	Hydraulic conductivity of blood vessel wall of normal and tumor tissue	$34.375 \times 10^{-14}\ \mathrm{m}^2/\mathrm{s}/\mathrm{kg}$ for normal tissue $3.5714 \times 10^{-10}\ \mathrm{m}^2/\mathrm{s}/\mathrm{kg}$ for tumor tissue	Equation below
K_T	Effectiveness of the treatment	$1.1 \cdot 10^{-3}\ \mathrm{g} \cdot \mathrm{cm}^{-3} \cdot \mathrm{day}^{-1}$	This study

Supplementary Table S2. Uncertainty parameters

Parameter	Description	Value	Reference
μ	shear modulus	Range for tumor (5-50 kPa)	[12–14]
k_{th}	hydraulic conductivity	Range for tumor ($K = \frac{2\mu(1+\nu)}{3(1-2\nu)}$ where Poisson ration ν= 0.4 and μ shear modulus)	[14]
S_v	Vascular density	Range for tumor (5000 m^{-1} -20000 m^{-1})	[15]

k_1	growth rate parameter	Range (0.1 - 0.3446 1/day)	[7]
p_v	Vascular pressure	Range for tumor (5-50 mmHg)	[9]
L_p	Hydraulic conductivity of blood vessel wall of normal and tumor tissue	Range for tumor (1.56x10^{-9} – 7.47x10^{-7} cm/Pa-s)	Equation below

Supplementary Table S3. Design parameters

Parameter	Description	Value	Reference
σ^f (sfT)	Reflection coefficient	In the range of (0.136 – 0.146) for tumor	Equations below
P_{er}	Vascular permeability	In the range of ($4.16 \cdot 10^{-9}$ m $\cdot\, s^{-1}$ – $6.225 \cdot 10^{-8}$) for tumor	Equations below
K_T(KT)	Effectiveness of the treatment	In the range of ($1.1 \cdot 10^{-4}$ – $1.1 \cdot 10^{-2}$ g $\cdot$ cm^{-3} $\cdot$ day^{-1})	This study
k_{on}(k_on)	Binding rate constant	In the range of ($1 - 1 \cdot 10^{6}$ cm$^{3} \cdot$ g$^{-1} \cdot$ day^{-1})	[8]
k_d	Blood circulation decay	In the range of (0.347 – 1.076 day)	[25]
t_0	Time of drug injection	In the range of (10 -14 day)	This study

P_{er} is the vascular permeability of the drug, L_p the hydraulic conductivity and σ_f the reflection coefficient. The parameters L_p, P_{er} and σ_f are expressed as a function of the vessel wall pores and the size of the conjugated-cytokines [26,27] :

$$L_p = \frac{\gamma r_0^2}{8\mu L_{vw}} \tag{1}$$

$$P_{er} = \frac{\gamma H D_0}{L_{vw}} \tag{2}$$

$$\sigma_f = 1 - W \tag{3}$$

where γ is the fraction of the vessel wall surface area occupied by pores, r_0 the pore radius, μ the viscosity and L_{vw} the thickness of the vessel wall. H and W describe the steric and hydrodynamic interactions of the conjugated-cytokines with the pores of the vessel wall that hinder diffusive and convective transport respectively and D_0 is the diffusion coefficient of a particle in free solution given by the Stokes-Einstein equation. By ignoring electrostatic interactions H and W become [27]:

$$H = \frac{6\pi F}{K_t} \tag{4}$$

$$W = \frac{F(2 - F)K_s}{2K_t} \tag{5}$$

where F is the partition coefficient expressed as:

$$F = (1 - \lambda)^2 \tag{6}$$

where λ is the ratio of the conjugated-cytokines size to the vessel wall pore size and K_t and K_s are expressed as [27]:

$$\binom{K_t}{K_s} = \frac{9}{4}\pi^2\sqrt{2}(1\text{-}\lambda)^{-5/2}\left[1 + \sum_{n=1}^{2}\binom{a_n}{b_n}(1\text{-}\lambda)^n\right] + \sum_{n=0}^{4}\binom{a_{n+3}}{b_{n+3}}\lambda^n. \tag{7}$$

Conclusions

In conclusion, the DCoMEX-BIO application successfully established a sophisticated AI-assisted multiphysics simulation-driven optimization framework aimed at enhancing the treatment protocols of immunotherapy. Through the development and refinement of two distinct versions of our mathematical model—first focusing on a five-equation framework without immune system considerations, and subsequently expanding it to a comprehensive twelve-equation model incorporating the dynamics of immune agents—we have significantly advanced our understanding of the tumor microenvironment and the interactions influencing treatment outcomes.

The introduction of surrogate models, including a hybrid approach that fuses computational-aided engineering (CAE) techniques with neural networks for time-history predictions and a dedicated neural network model for discrete time-point evaluations, has proven essential in managing the extensive computational demands associated with Bayesian updates. This innovative strategy has enabled us to accurately capture the time evolution of tumor volume, a critical indicator of immunotherapy efficacy, while addressing the inherent uncertainties in model parameterization.

Furthermore, our brute-force optimization analysis of immunotherapy injection schedules has yielded valuable insights into the optimal timing strategies, demonstrating the framework's practical applicability in clinical settings. The Bayesian inference procedure effectively utilized experimental data to refine model parameters, enhancing the robustness and reliability of our predictions.

The sources of experimental data, crucial for model validation and updates, have been meticulously described, ensuring transparency and reproducibility in our methodologies. Overall, the integration of advanced mathematical modeling, surrogate approaches, and optimization techniques within this framework offers a significant contribution to the field of cancer immunotherapy, paving the way for more personalized and effective treatment strategies tailored to individual patient needs. The outcomes of this work lay a strong foundation for future research endeavors, potentially guiding the clinical application of immunotherapy protocols in real-world settings.

References

References

1. Roose T, Netti PA, Munn LL, Boucher Y, Jain RK. Solid stress generated by spheroid growth estimated using a linear poroelasticity model. Microvascular Research. 2003;66: 204–212. doi:10.1016/S0026-2862(03)00057-8

2. Kim Y, Stolarska MA, Othmer HG. The role of the microenvironment in tumor growth and invasion. Progress in Biophysics and Molecular Biology. 2011;106: 353–379. doi:10.1016/j.pbiomolbio.2011.06.006

3. MacLaurin J, Chapman J, Jones GW, Roose T. The buckling of capillaries in solid tumours. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences. 2012;468: 4123–4145. doi:10.1098/rspa.2012.0418

4. Voutouri C, Stylianopoulos T. Evolution of osmotic pressure in solid tumors. Journal of Biomechanics. 2014;47: 3441–3447. doi:10.1016/j.jbiomech.2014.09.019

5. Goldsworthy JD, editor. Interpreting constitutions: a comparative study. 1. publ. in paperback. Oxford: Oxford Univ. Press; 2007.

6. Stylianopoulos T, Martin JD, Snuderl M, Mpekris F, Jain SR, Jain RK. Coevolution of Solid Stress and Interstitial Fluid Pressure in Tumors During Progression: Implications for Vascular Collapse. Cancer Research. 2013;73: 3833–3841. doi:10.1158/0008-5472.CAN-12-4521

7. Harkos C, Stylianopoulos T, Jain RK. Mathematical modeling of intratumoral immunotherapy yields strategies to improve the treatment outcomes. PLoS Computational Biology. 2023;19: e1011740. doi:10.1371/journal.pcbi.1011740

8. Harkos C, Stylianopoulos T. Investigating the synergistic effects of immunotherapy and normalization treatment in modulating tumor microenvironment and enhancing treatment efficacy. Journal of Theoretical Biology. 2024; 111768. doi:10.1016/j.jtbi.2024.111768

9. Baxter LT, Jain RK. Transport of fluid and macromolecules in tumors. I. Role of interstitial pressure and convection. Microvascular Research. 1989;37: 77–104. doi:10.1016/0026-2862(89)90074-5

10. Mow VC, Kuei SC, Lai WM, Armstrong CG. Biphasic Creep and Stress Relaxation of Articular Cartilage in Compression: Theory and Experiments. Journal of Biomechanical Engineering. 1980;102: 73–84. doi:10.1115/1.3138202

11. Taber LA. Theoretical study of Beloussov's hyper-restoration hypothesis for mechanical regulation of morphogenesis. Biomech Model Mechanobiol. 2008;7: 427–441. doi:10.1007/s10237-007-0106-x

12. Eder M, Raith S, Jalali J, Volf A, Settles M, Machens H-G, et al. Comparison of Different Material Models to Simulate 3-D Breast Deformations Using Finite Element Analysis. Annals of Biomedical Engineering. 2014;42: 843–857. doi:10.1007/s10439-013-0962-8

13. Samani A, Zubovits J, Plewes D. Elastic moduli of normal and pathological human breast tissues: an inversion-technique-based investigation of 169 samples. Physics in Medicine and Biology. 2007;52: 1565–1576. doi:10.1088/0031-9155/52/6/002

14. Netti PA, Berk DA, Swartz MA, Grodzinsky AJ, Jain RK. Role of Extracellular Matrix Assembly in Interstitial Transport in Solid Tumors. Cancer Research. 2000;60: 2497–2503.

15. Harkos C, Svensson SF, Emblem KE, Stylianopoulos T. Inducing Biomechanical Heterogeneity in Brain Tumor Modeling by MR Elastography: Effects on Tumor Growth, Vascular Density and Delivery of Therapeutics. Cancers. 2022;14: 884. doi:10.3390/cancers14040884

16. Casciari JJ, Sotirchos SV, Sutherland RM. Variations in tumor cell growth rates and metabolism with oxygen concentration, glucose concentration, and extracellular pH. Journal of Cellular Physiology. 1992;151: 386–394. doi:10.1002/jcp.1041510220

17. Mpekris F, Angeli S, Pirentis AP, Stylianopoulos T. Stress-mediated progression of solid tumors: effect of mechanical stress on tissue oxygenation, cancer cell proliferation, and drug delivery. Biomechanics and Modeling in Mechanobiology. 2015;14: 1391–1402. doi:10.1007/s10237-015-0682-0

18. Stylianopoulos T, Martin JD, Snuderl M, Mpekris F, Jain SR, Jain RK. Coevolution of solid stress and interstitial fluid pressure in tumors during progression: Implications for vascular collapse. Cancer Research. 2013;73: 3833–3841. doi:10.1158/0008-5472.CAN-12-4521

19. Friedman A, Hao W. The Role of Exosomes in Pancreatic Cancer Microenvironment. Bulletin of Mathematical Biology. 2018;80: 1111–1133. doi:10.1007/s11538-017-0254-9

20. Voutouri C, Kirkpatrick ND, Chung E, Mpekris F, Baish JW, Munn LL, et al. Experimental and computational analyses reveal dynamics of tumor vessel cooption and optimal treatment strategies. Proceedings of the National Academy of Sciences. 2019;116: 2662–2671. doi:10.1073/pnas.1818322116

21. Lai X, Friedman A. Combination therapy of cancer with cancer vaccine and immune checkpoint inhibitors: A mathematical model. PLoS ONE. 2017;12: e0178479. doi:10.1371/journal.pone.0178479

22. Schmidt MM, Wittrup KD. A modeling analysis of the effects of molecular size and binding affinity on tumor targeting. Molecular Cancer Therapeutics. 2009;8: 2861–2871. doi:10.1158/1535-7163.MCT-09-0195

23. Mok W, Stylianopoulos T, Boucher Y, Jain RK. Mathematical modeling of herpes simplex virus distribution in solid tumors: Implications for cancer gene therapy. Clinical Cancer Research. 2009;15: 2352–2360. doi:10.1158/1078-0432.CCR-08-2082

24. Pluen A, Boucher Y, Ramanujan S, McKee TD, Gohongi T, Di Tomaso E, et al. Role of tumor-host interactions in interstitial diffusion of macromolecules: Cranial vs. subcutaneous tumors. Proceedings of the National Academy of Sciences of the United States of America. 2001;98: 4628–4633. doi:10.1073/pnas.081626898

25. Chauhan VP, Stylianopoulos T, Martin JD, Popović Z, Chen O, Kamoun WS, et al. Normalization of tumour blood vessels improves the delivery of nanomedicines in a size-dependent manner. Nature Nanotechnology. 2012;7: 383–388. doi:10.1038/nnano.2012.45

26. Mpekris F, Angeli S, Pirentis AP, Stylianopoulos T. Stress-mediated progression of solid tumors: effect of mechanical stress on tissue oxygenation, cancer cell proliferation, and drug delivery. Biomechanics and Modeling in Mechanobiology. 2015;14: 1391–1402. doi:10.1007/s10237-015-0682-0

27. Deen WM. Hindered transport of large molecules in liquid-filled pores. AIChE Journal. 1987;33: 1409–1425. doi:10.1002/aic.690330902