

Alma Mater Studiorum - Università di Bologna

DOTTORATO DI RICERCA IN  
SCIENZE E TECNOLOGIE DELLA SALUTE

Ciclo 35

**Settore Concorsuale:** 09/G2 - BIOINGEGNERIA

**Settore Scientifico Disciplinare:** ING-IND/34 - BIOINGEGNERIA INDUSTRIALE

ORCHESTRATION OF MULTISCALE MODELS FOR COMPUTATIONAL  
ONCOLOGY

**Presentata da:** Vinícius Clemente Varela

**Coordinatore Dottorato**

Marco Viceconti

**Supervisore**

Marco Viceconti

**Co-supervisore**

Luis Martì Bonmatì

**Esame finale anno 2023**

I first thank UNIBO and PRIMAGE for the opportunity and funding for the entire period of the doctorate.

To my parents and sisters, for the encouragement, for the investment in my education, and the support for this life change

To Professor Marco Viceconti the supervision, his teachings, and, his honest feedbacks. I learned and changed a lot from you and from observing you.

To Professor Barbara Quintela for her supervision, patience (or attempt to) and, willingness to help. Especially for her essential support in the first half of the doctorate.

To Doctor Luis Martì-Bonmatì for supervision and motivation.

I heartfully thank all the friends and colleagues I met during those three years in Bologna.

Highlighting the friends from the lab with whose I had more contact. In alphabetical order, I thank Alessandra, Alex, Andrea, Chiara, Chloe, Cristina, Domitille, Francesca, Ferninando, Giacomo, Giorgio, Goran, Julia, Maila, Nino, Roberta, Samuele, Sara, Sofia, Sylwia, Valentina, and those I am struggling to remember now or met recently.

You all changed my perception of the world in some way and at some point helped me to deal with the solitude of living abroad.

Last but not least, I would like to thank Italy and the Italians for being a country so open to foreigners (despite the bureaucracy) and so receptive and cozy most at time. And for the great food too.



## **Abstract**

Cancer is a challenging disease that involves multiple types of biological interactions in different time and space scales. Often computational modelling has been facing problems that, in the current technology level, is impracticable to represent in a single space-time continuum. To handle this sort of problems, complex orchestrations of multiscale models is frequently done. PRIMAGE is a large EU project that aims to support personalized childhood cancer diagnosis and prognosis. The goal is to do so predicting the growth of the solid tumour using multiscale *in-silico* technologies. The project proposes an open cloud-based platform to support decision making in the clinical management of paediatric cancers. The orchestration of predictive models is in general complex and would require a software framework that support and facilitate such task. The present work, proposes the development of an updated framework, referred herein as the VPH-HFv3, as a part of the PRIMAGE project. This framework, a complete re-writing with respect to the previous versions, aims to orchestrate several models, which are in concurrent development, using an architecture as simple as possible, easy to maintain and with high reusability. This sort of problem generally requires unfeasible execution times. To overcome this problem was developed a strategy of particularisation, which maps the upper-scale model results into a smaller number and homogenisation which does the inverse way and analysed the accuracy of this approach.

## **Index**

<b>1. Introduction .....</b>	<b>8</b>
1.1. <i>The clinical problem</i> .....	8
1.1.1. Neuroblastoma.....	9
1.1.2. Symptoms .....	10
1.1.3. Diagnosis .....	10
1.1.4. Treatment.....	11
1.1.5. Prevention.....	12
1.2. <i>The technical problem</i> .....	14
1.3. <i>The PRIMAGE research project</i> .....	15
1.3.1. Partnership.....	16
1.3.2. Infrastructure.....	16
1.3.3. In-silico scale models .....	16
1.3.4. Pre-processing scripts .....	17
<b>2. The PRIMAGE multiscale model .....</b>	<b>19</b>
2.1. <i>Scale separation analysis</i> .....	19
2.1.1. The idealisation of tumour growth: a mathematical model.....	19
2.1.2. The problem of scale separation .....	20
2.1.3. Scale separation analysis for the Neuroblastoma model .....	21
2.2. <i>The orchestration target: the PRIMAGE model</i> .....	23
2.2.1. The inputs .....	23
2.2.2. The component models.....	24
2.2.3. The outputs .....	27
<b>3. The orchestration framework: VPH-HF v3.....</b>	<b>28</b>
3.1. <i>Introduction</i> .....	28
3.1.1. The need for models' orchestration .....	28
3.1.2. A critical review of existing orchestration frameworks .....	28
3.1.3. The High-Level Architecture for distributed simulation .....	29
3.2. <i>The VPH-HF v3 architecture</i> .....	31
3.2.1. Overview.....	32
3.2.2. Database.....	34
3.2.3. Master & Commander .....	37
3.2.4. Serialisation .....	38
3.2.5. De-serialisation .....	39
3.2.6. Relation modules .....	42
3.2.7. Implementation details.....	43
<b>4. Development and validation of a particularization strategy .....</b>	<b>44</b>
<b>EFFECT OF PARTICULARISATION SIZE ON THE ACCURACY AND EFFICIENCY OF A MULTISCALE TUMOURS' GROWTH MODEL .....</b>	<b>44</b>
<i>ABSTRACT</i> .....	44
<i>KEYWORDS</i> .....	44
4.1. <i>INTRODUCTION</i> .....	44
4.2. <i>MATERIALS AND METHODS</i> .....	46
4.2.1. A Brief overview of the biological problem and its idealisation.....	46
4.2.2. Implementation details.....	47
4.2.3. Data.....	48

4.2.4.	The particularisation/homogenisation relation model .....	48
4.2.5.	The validation study .....	49
4.3.	<i>RESULTS</i> .....	49
4.4.	<i>DISCUSSION</i> .....	52
<b>5.</b>	<b>The PRIMAGE orchestration .....</b>	<b>55</b>
5.1.	<i>Description of the final implementation</i> .....	55
5.1.1.	The component models and the orchestrator layer .....	55
5.1.2.	The HPC platform.....	57
5.2.	<i>Execution profiling for the multiscale model</i> .....	58
5.3.	<i>Scalability and convergence of the particularisation algorithm</i> .....	62
5.4.	<i>Selected results</i> .....	65
5.5.	<i>Conclusions and future works</i> .....	68
<b>6.</b>	<b>References.....</b>	<b>69</b>

## **ACRONYMS**

ABM – Agent Based Model

AIOM – Italian Foundation of Medical Oncology

API – Application Programming Interface

CT – Computer Tomography

DWI – Diffusion-Weighted Image

DCE – Dynamic Contrast-Enhanced

FEM – Finite Element Model

HPC – High-Performance Computer

INRGSS – The International Neuroblastoma Risk Group Staging System

INSS – International Neuroblastoma Staging System

MRI – Magnetic Resonance Imaging

MIBG - Iodine-linked metaiodobenzylguanidine

RVE – Reference Volume Element

# 1. Introduction

## 1.1. The clinical problem

Every year, millions of new cases of cancer are registered worldwide. In the past decades, cancer has been the second cause of death worldwide, only after cardiovascular diseases (Ritchie et al., 2018). In 2020 the world incidence rate was 201 per 100.000 inhabitants representing more than 19 million new cases (World Health Organization, 2021a)

Cancer is a group of diseases in which some cells with unhealthy mutations multiplies uncontrollably. It is a result of chance and the human body's regeneration processes. Human cells are constantly growing and multiplying and forming new cells to meet the body's needs (National Institute of Cancer, 2021). Errors during cell division cause mutations, which are replicated in all cells that derive from the same replication line.

In 2000, a popular work defined the "Hallmarks of Cancer" as a set of criteria to distinguish cancer cells from normal cells (Hanahan and Weinberg, 2000). The authors define a list of functional capabilities acquired by the human cells that are crucial for the development of malignant tumours: the lack of external stimulation to keep growing, the loss of sensibility to signals to prevent growing, resistance to apoptosis, the capability of dividing continuously, the vascularization stimulation to bring blood to the tumour and the active invasion to surrounding tissues and metastasis. Years later two characteristics were added to the Hallmarks, the dysregulation of cellular metabolism, and the evasion of the immunity system (Hanahan and Weinberg, 2011).

The human body is composed of trillions of cells of different types, but a tumour may start from a single cell. As a tumour gets bigger, cancer can spread to the tissues nearby, secrete enzymes that break normal cells and tissues, and release in the vascular and lymphatic systems cancer cells that cause metastases, secondary tumours that form across the body (fig. 1.1) (Canadian Cancer Society, n.d.). It is known that those cell alterations are caused by the sum of genetic factors, physical factors (e.g., radiation), chemical factors (e.g., alcohol, tobacco, air contamination, substances in processed foods, etc.), and biological factors (e.g., virus and bacteria) .

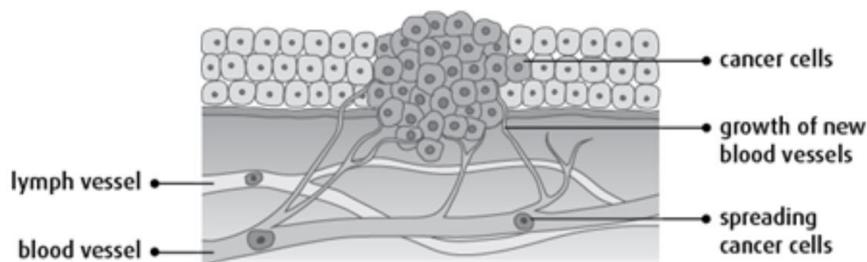


FIGURE 1.1: Cancer spreading (Canadian Cancer Society, 2016)

All of these factors can increase the probability of mutations and consequently increase the probability of errors that occur in sensible parts of the genome. These mutations can be dangerous when they involve important genes that control the cells. If due to a mutation one cell loses a chromosome (aneuploidy), which is a piece of major genetic information, all the cells created by this one will have one chromosome less, missing some information. In case of losing, for example, the parts that control cell growth, cells can grow uncontrollably and form tumours (Matthay et al., 2016).

Time is also an aspect that can impact the number of mutations, as also aging can lead to loss of efficiency at the cell reparation mechanisms. This is the reason cancer is rarer in children. The Italian foundation of medical oncology (AIOM) report from 2020 states that childhood tumours (0-14 years old) represent less than 0.5% of the tumors' cases, with a rate of only a few dozen

cases per 100.000 inhabitants in Italy (Fig 1.2) (Associazione Italiana di Oncologia Medica et al., 2020). In developing countries, where the children population is bigger and there is less access to better diagnosis, this rate varies between 3% to 10%. Worldwide nearly 400.000 cases of pediatric cancer are estimated yearly (World Health Organization, 2021b).

Pediatric cancers are different from those types of cancers developed in adults. There are few known risk factors for cancer occurring in children or adolescents. Exposure to ionizing radiation may increase the risk of leukemia, brain tumours, and possibly other neoplasms. The use of drugs that suppress the immune system applied for solid organ transplant recipients might increase the risk of lymphoma, as well as infection with the Epstein-Barr virus (American Cancer Society, 2022). However, in children, cancer prominently originates from epigenetic dysregulations.

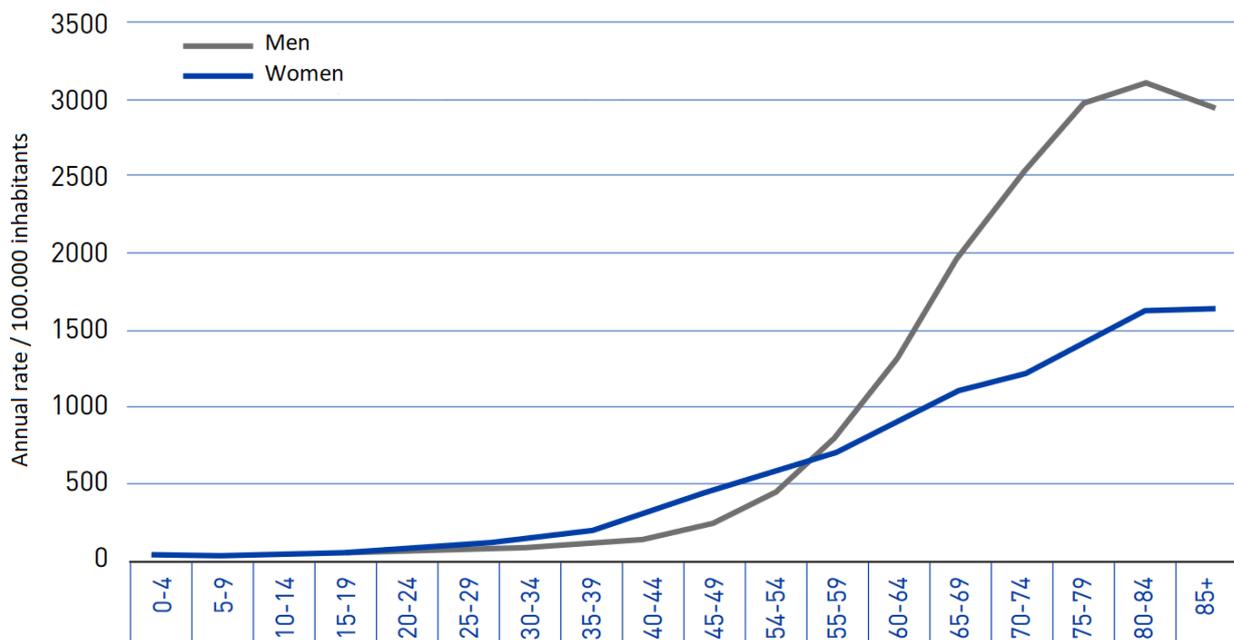


FIGURE 1.2: Annual rate by 100.000 inhabitants of tumour cases in Italy by age range for male (blue) and female (black).

### 1.1.1. Neuroblastoma

Neuroblastoma is the most frequent solid tumour in children under 1 year of age, classified in the group of sympathetic nervous system tumours that represents 7.8% of cancer cases in the worldwide population below 15 years old. Historically neuroblastoma and central neural system tumours are the second most prevalent pediatric cancer, while neuroblastoma is the most incidental tumour among the population up to 1-year-old (Hung et al., 2015; Santos, 2019; World Health Organization, 2021b). However, it is not a high-prevalent disease. In Europe, it was diagnosed 10.9 cases per million between 1988-1997 (Spix et al., 2006), in Taiwan 41.23 cases per million were registered in the period 1995-2014 (Hung et al., 2015), in Brazil neuroblastoma represents about 4.6% of childhood tumours which is equal to 11.5 cases per million children. Among children, up to 1-year-old, these rates increase such as 243 per million in France (2000-2004) and 255 in the United States (2006-2010) (Santos, 2019).

Neuroblastoma takes place on the neuroblasts, embryonic cells precursor of the nerve cells that live in the portion of the nervous system, which is responsible for involuntary actions such as digestion, heart beating, and breathing. The probability of developing the disease can be inherited from the parents since the neuroblasts are developed during pregnancy, but commonly it is associated with modifications in the genes that can happen spontaneously or for natural

reasons during the development stages of a child (Associazione Italiana per la Ricerca sul Cancro, 2019).

The term *neuroblastoma* includes several cancer diseases that originated on the neuroblasts. They differ based on the composition of the cells and their differentiation. Differentiation is the process of maturation of cells wherein immature cells differentiate until reach a mature form with specialized functions. The differentiation degree of neuroblasts and Schwann cells obtained on how cells look under a microscope determines which type of tumour is being generated. Ganglioneuroma is the benign tumour, neuroblastoma is the malignant form and ganglioneuroblastoma is the intermediate state, it has characteristics of the benign and the malignant tumours. Benign tumours are less probably to spread while malignant tumours are aggressive regarding growth and diffusion (Bayeva et al., 2021).

### 1.1.2. Symptoms

Most common in children under 5 years old, neuroblastoma usually appears in the abdomen, especially in the adrenal gland, causing swelling appearance and can also be accompanied by loss of appetite (fig. 1.3) (American Cancer Society, 2022).



FIGURE 1.3: Main symptoms of Neuroblastoma (American Childhood Cancer Organization, 2018).

However, the symptoms and signals can depend on the location of the tumour. Neuroblastoma can arise in the adrenal gland, in the cervical, thoracic, abdominal, and/or pelvic regions, following the course of the entire sympathetic nervous system chain. Often it can present symptoms like tachycardia, hypertension, and fever (Associazione Italiana per la Ricerca sul Cancro, 2019).

The immune system is able, through several physiological mechanisms, of recognizing and destroying mutated cells in addition to pathogens. However, tumour cells, including neuroblastoma cells, acquire as part of their mutation the ability to evade detection by the immune system (Raffaghello et al., 2005) In addition, cancer can weaken the immune system, especially when it spreads in the bone marrow, where important blood cells for immunity are produced (Cancer Research UK, 2014; Matthay et al., 2016).

### 1.1.3. Diagnosis

Besides the identification of the symptoms, the doctor also investigates the hereditary medical history before asking for exams. Then, blood and urine tests are requested focusing on the urinary catecholamine and its derivatives which are produced by nervous cells. In case of suspicion, imaging

diagnostic confirms the appearance of the disease. Typically, computerized tomography (CT) or magnetic resonance imaging (MRI) in the region of interest gives the information necessary to understand the progress of the tumour (American Cancer Society, 2021a). It is also possible to detect neuroblastomas during pregnancy through ultrasound techniques (Granata et al., 2000).

Having the diagnosis confirmed, the next key step is to identify the stage of the disease. The CT and MRI are also used for this step although, the biopsy, that is, the removal of a sample of the tumour cells followed by the respective analysis is the most common way to confirm the neuroblastoma diagnosis. It is when is possible to define the type and the differentiation degree of the tumour. Finally, an analysis of scintigraphy imaging using radioactive iodine-linked metaiodobenzylguanidine (MIBG) can define with high accuracy level how much and in which areas the tumour has spread and obtain indispensable information about the response to treatment (Associazione Italiana per la Ricerca sul Cancro, 2019).

The detection of the tumour grade is crucial to determine the therapy treatment. Currently, there are two systems to determine the child’s risk group before treatment. The International Neuroblastoma Risk Group Staging System (INRGSS) uses results from imaging tests, while the INSS (International Neuroblastoma Staging System) takes the result of the surgical biopsy to classify into 4 stages, depending on how the spread is the neuroblastoma, with stage 4 being the riskiest stage (American Cancer Society, 2021b). In the Netherlands, most of the cases are high-risk at the diagnostic moment (Fig. 1.4) (Tas et al., 2020).

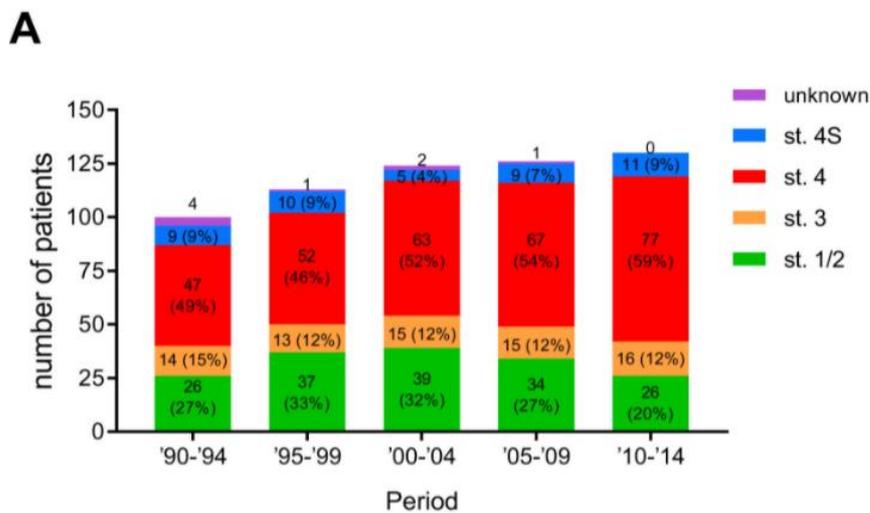


FIGURE 1.4: Number of diagnosis by stage for each 5-year interval (Tas et al., 2020).

#### 1.1.4. Treatment

A precise diagnosis is essential to an efficient treatment as far as each type of cancer can request different approaches such as surgery, chemotherapy, and radiotherapy. In some cases, palliative care can also be applied in order to attenuate the symptoms of the disease and bring more life quality to the patient (Cancer.net, 2012; Organização Pan-Americana da Saúde, 2020).

Basically, the surgical removal of the tumor is the first line of treatment, but it depends on the dimension and location of the tumour. Typically, the surgery is applied in the early stages of the disease or at the end of some treatments when the tumour has shrunk. However, this approach can be also applied in cases where the neuroblastoma has spread to remove as much tumour as possible (Cancer.net, 2012). When the tumour is large, and consequently, not removable, chemotherapy is the most indicated primary treatment. Chemotherapy involves the administration of a set of drugs that aims to destroy the cancer cells preventing them from growing, dividing, and multiplying. The side effects of the chemotherapy depend on the child, the dosage and how long is the treatment. They

usually include hair loss, vomiting, diarrhea, loss of appetite etc. They disappear once the treatment is finished.

High-risk patients are subject to radiotherapy, which uses radiating energy to destroy cancer cells. Also, this treatment may involve several side effects, and some of them can persist after the end of the treatment, such as problems in the development of brain, ovaries, and testicles (Cancer.net, 2012; Organização Pan-Americana da Saúde, 2020).

Nowadays immunotherapy has presenting great results on neuroblastoma treatment. It increased the 5-year survival rates of patients with neuroblastoma (Matthay et al., 2016). This technique aims to provide to stimulate the immune system to acquire an immunological memory against the tumour. The state of the art of treatments that work in adults cannot be applied to paediatric cancer given the low immunogenicity of the tumour although progress has been done in this approach to treat neuroblastoma tumours (Wienke et al., 2021).

In some cases, the treatment can be combined with autologous hematopoietic stem cell transplantation. These cells are collected from the patient's bloodstream and during high-dose chemotherapy are infused back in order to replace blood stem cells to create healthy bone marrow which is partially destroyed by chemotherapy. This is a high-risk treatment and is used only for very advanced stages of the disease. The care plan always includes treatments for the side effects and children in advanced cases are also subjected to treatment after the cure to prevent relapse (Associazione Italiana per la Ricerca sul Cancro, 2019).

Besides the side effects, neuroblastoma treatments cause emotional, social, and financial effects as well. Palliative care is also part of the treatment and includes caring for how children feel during the treatment, family support, and other non-medical needs. It might involve medical, diet changes, relaxation techniques, and spiritual, emotional or even religious support.

Significant advances have been made on cancer management. The improvements of all the technologies for treatments against cancer in recent decades caused a reduction in the death rates for all types of neuroblastoma (Children's Cancer and Leukemia Group, 2019). Due to the advances in technology to diagnose and treat cancer, nowadays about 80% of children with cancer survive 5 years or more if early diagnosed and receive the correct treatment. This represents an increase of 22% in comparison to 1970 (58%) (Oncoguia, 2020). The death rate for childhood cancer has decreased from 6.3 per 100,000 to 1.8 in the period from 1970 to 2019 in the United States of America, especially due to improvements in treatment and high participation in clinical trials (American Cancer Society, 2022).

#### *1.1.5. Prevention*

Around 40% of cancer cases could be prevented by tackling risk factors relating to lifestyle choices. Nowadays it is known that to have a healthy weight, be physically active, have a good diet, not smoke, limit the exposition to ultraviolet radiation, be vaccinated against some viruses, and control the consumption of sugar, red meat, alcohol, and processed foods can make an enormous impact on people's likelihood of developing cancer (World Cancer Research Fund International, 2022).

Unlike adult cancers, in the current state of knowledge, most childhood cancers cannot be prevented, as lifestyle factors might take many years to influence cancer risk (American Cancer Society, 2021) (AIRC, 2019). The disease arises during the growth and development of cells, which makes strategies such as avoiding delayed diagnoses, assurance of access to high-quality treatment, and improvement to treatments the best approaches to reduce the death rate (World Health Organization, 2021b). Whilst psychosomatics medicine has been researching if emotional and psychological aspects can influence the appearance of tumours (Gostoli et al., 2021; Hitzer and León-Sanz, 2016; Zander, 1983), it is still discussed if cancer is a psychosomatic disease or not. What is accepted is that the disease arises during growth and development of embryonic cells which make strategies such as avoiding delayed diagnoses, assurance of access to high quality treatment and

improvement to treatment technologies the best approaches to reduce the death rate (World Health Organization, 2021b).

Cancer development can be reduced and controlled if the diagnostic is done at the early stages of the disease. It is estimated that less than 5 out of 100 children with low-risk neuroblastoma survive for less than 5 years long, while for high-risk neuroblastoma the death in the same period can reach 50% of the cases (Cancer.net, 2012; Siegel et al., 2021). A lot of cancers have better probability of cure if precociously detected and treated correctly. Fig. 1.5 shows the survival rate in Netherlands for each stage of the disease where gets explicit the diagnosis stage implies significantly at the likelihood of cure (Tas et al., 2020).

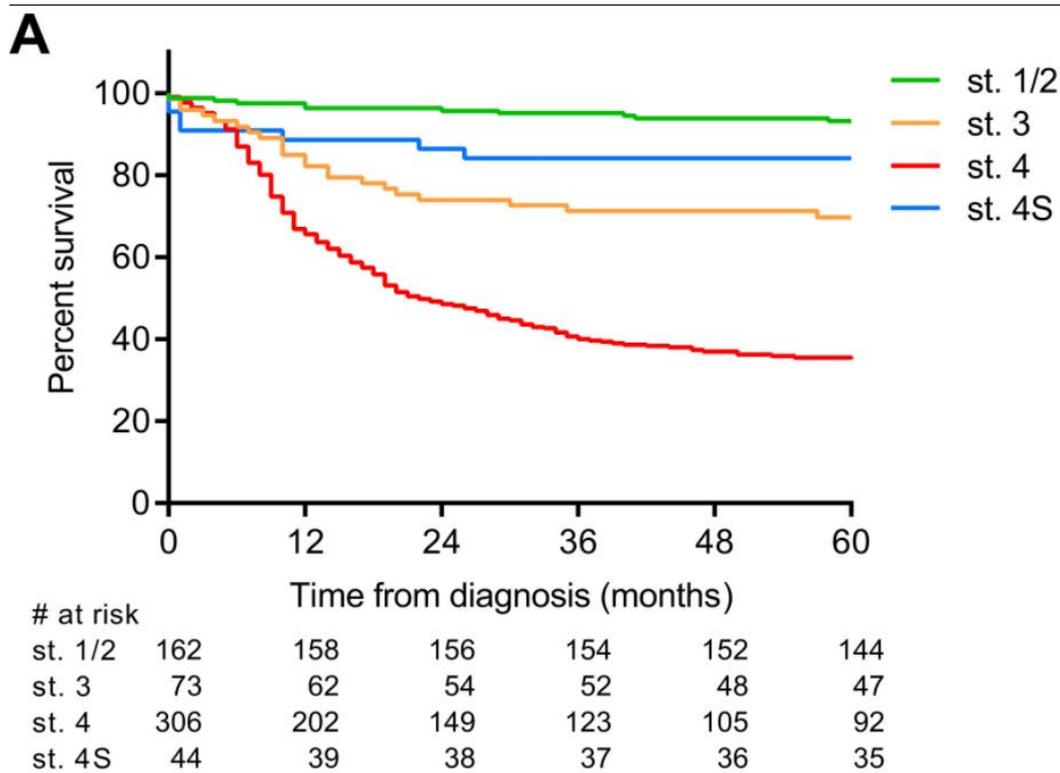


FIGURE 1.5: Survival rate of different neuroblastoma stages at the diagnostic of history data in Netherlands (Tas et al., 2020). Stage 1 represents the low-risk cases and stage 4 the high-risk. The stage 4S represents a special type of 4 stage neuroblastoma.

The probability of a childhood cancer survival is also highly associated to the socioeconomic conditions of the child’s family, low-income countries have the worst survival rates (Fig 1.6) (Ward et al., 2019). Treatment access still varies widely between countries and within a country. The report of the 2019 global survey reported that 29% of low-income countries have medicines for cancer chemotherapy available for their populations while this number is equal to 96% in high-income countries (World Health Organization, 2020a). Despite of increasing the amount of early diagnosis, the global action plan of United Nations together with the World Health Organization commits to incrementally build efficient and affordable cancer management systems, involving all levels of health care system such as medical oncology, surgery, radiation oncology, pathology, palliative care, psycho-oncology, oncology nursing, nutrition and rehabilitation (World Health Organization, 2022).

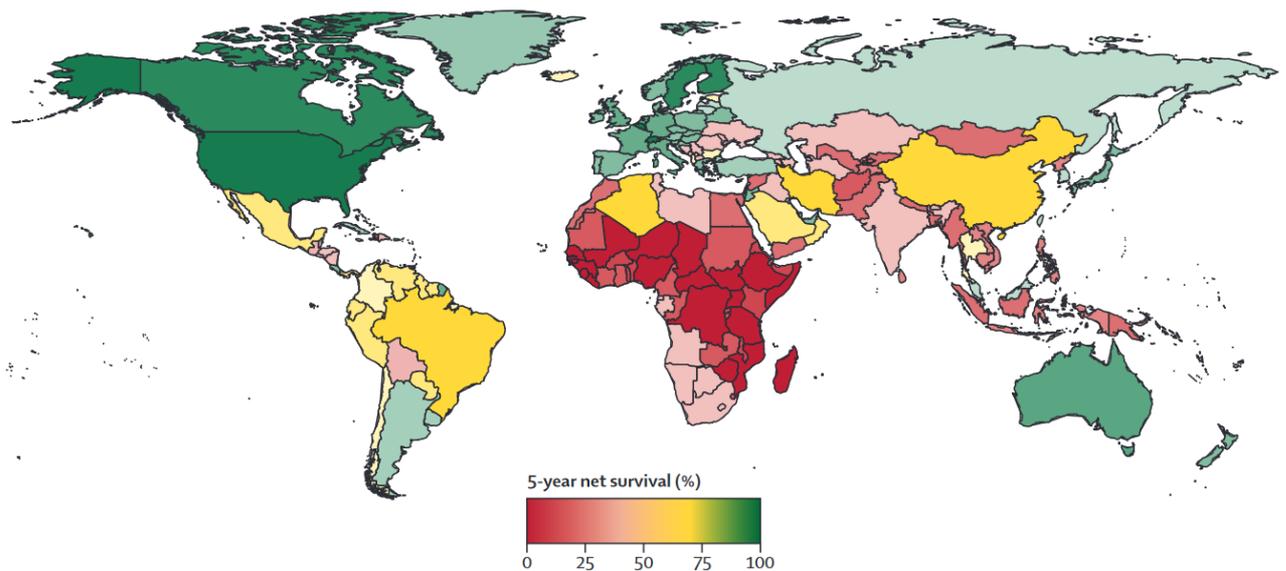


FIGURE 1.6: Estimated childhood cancer 5-year net survival by country (2015–2019) (Ward et al., 2019).

The incidence of cancer diseases has been burdening the entire world and with the knowledge currently available less than 50% of cancers cases are preventable, representing one of the major barriers to development in the 21<sup>st</sup> century (World Health Organization, 2020b). Given that, governments around the world committed themselves to controlling and preventing cancer. Consequently, great finances are allocated to research and innovation on these diseases in order to achieve better and/or cheaper treatments.

## 1.2. The technical problem

Since there are various possible treatments for neuroblastoma, it would be very useful to know before the treatment starts how each of the available treatment options would affect the volume of the tumour.

Modelling the change of volume of a cancer tumour as a function of the treatment, with all the complexity this sort of application imposes, is a difficult task. The phenomena involved require such a large range of temporal and spatial dimensions. To predict the growth of a tumour based on the treatment, it is necessary to model physical behaviours from the tumour scale to the effects of the medicine at a molecular level passing through the tissue scale for simulating the cell interactions. Therefore, we have very small biological entities producing effects that manifest at much higher spatial scales (e.g., tissue, organ, organism). Regarding the temporal scale, those biological short-time reactions can affect observable consequences for months or even years.

A neuroblastoma tumour can be as big as some centimetres, while the subcellular reactions involve proteins in the order of nanometers. At the temporal scale, a neuroblastoma treatment can last for months while these chemical reactions can happen as fast as 10 milliseconds. Given that, it is impossible to represent this problem mathematically in a continuum of space and time scales, we are forced to work on a multiscale modelling approach which decomposes the problem into smaller ones, each one defined at a particular space-time scale representing the phenomenon of interest at the respective scale range.

Multiscale modelling is a popular approach to overcome this sort of problem. It consists of decomposing a large-scale problem in multiple space-time partitions and defining a model to represent the phenomenon of interest inside the respective scale (Fig 1.7). Then all these parts must to be connected by an orchestrator layer.

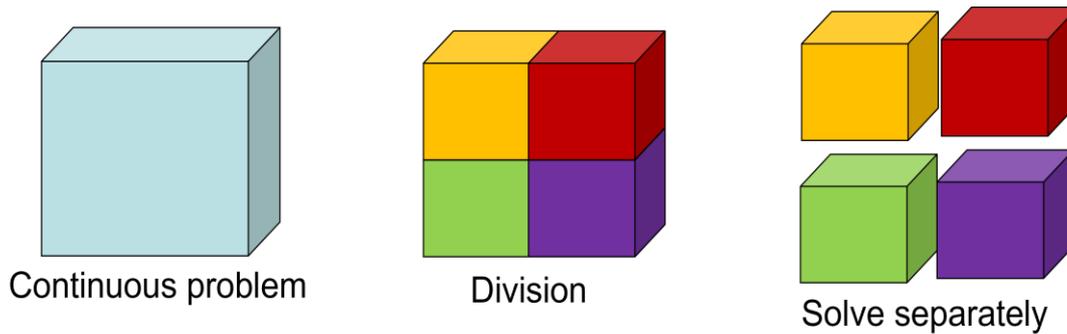


FIGURE 1.7: Multiscale modelling approach

The orchestration of predictive models is in general complex and would require a software framework to support and facilitate such task. The orchestrator is a framework that must ensure centralised management of the *control flow* (e.g., the set of executions instructions to be imparted to each model during the orchestration), the *data flow*, and the necessary handling and transformation of data between the output of a model and the input of another.

We also face computational limitations. For example, a Reference Volume Element (RVE) of volume equal to  $2e^{-9} \text{ m}^3$  takes 100 seconds to simulate a 2-week simulation and the whole primary tumour volume in the order of  $1e^{-3} \text{ m}^3$ . It would be necessary to implement 500 thousand RVE to solve this model at a continuous scale. This would take 5,787 days of execution days in a one CPU core simulation. Even if we did speed up the model a couple of times the execution time would remain unfeasible for a real end application.

Dividing the problem does not imply reducing the computational cost. And as solving this model exceeds the computational capability that is available, an additional approach must be applied. The computational cost can be reduced to reasonable levels if instead of calculating all the RVE, we estimate the variables for a part of them. One way of doing it is interpolating the values of the variables based on values obtained by running the models for a sample of RVE. This can be done by coupling particularization and homogenization methods into the orchestrator, whereas the particularization is responsible for defining the samples.

The aim of this research project is the development of a multiscale orchestration software framework, referred to hereinafter as VPH-HF v3, which includes also a validated method for particularization and homogenisation, to make the computational cost acceptable. The framework was developed as part of the EU-funded PRIMAGE project, which aims to develop an *in-silico* treatment optimization tool for neuroblastoma. VPH-HF v3 is the third attempt to develop a multiscale orchestration framework that executes with little computational overhead, and it is easy to maintain and further develop to fit additional specifications. The main assumption of the project is that these orchestrations involve weakly coupled models, for which the management of the data flow is prevalent.

### 1.3. The PRIMAGE research project

PRIMAGE is a large EU project that aims to develop an open cloud-based platform for supporting oncology decision-making in clinical treatments on children (Martí-Bonmatí et al., 2020). This platform will offer clinical assistance for diagnosis, treatment allocation and prognosis. The project focuses on the development of *in silico* tools for personalized clinical management of childhood cancers considering the progression of the tumour from the diagnosis. The decision support platform is being constructed using imaging biomarkers and *in silico* medicine research and is validated on neuroblastoma tumours.

### 1.3.1. Partnership

Interdisciplinarity is a key aspect of PRIMAGE consortium, which is composed of a very heterogeneous group of private and public organizations, to perform this collaborative research. The project requires a research workforce with knowledge in infrastructure, biobanks, oncology, computational modelling and cloud computing.

The work presented within this thesis includes the design and development of a framework to orchestrate the simulation of models developed by other partners. Three institutions given by the Universidad the Zaragoza (UNIZAR), Chemotargets SME (CHMT) and the University of Sheffield (USFD) worked in the development of four models at different space-time scales to simulate the tumour volume growth with or without considering the effect of therapy.

The framework developed must run remotely in an HPC system, which was done in collaboration with Cyfronet which offered two clusters (Prometheus and Ares) throughout the project and supported all the development of the tumour multiscale model.

Several other partners take part in PRIMAGE with a minor connection to this work.

### 1.3.2. Infrastructure

The resources to PRIMAGE platform include a large-scale HPC system (with high-performance CPUs and GPUs) overlaid by a data access PL-grid data and the Rimrock API controller. Also, hybrid cloud resources will host the PRIMAGE data repositories and an integration middleware consisting of public and private cloud computing and storage, a set of protocols and interfaces between HPC/storage, and external data sources composed of pseudo-anonymized clinical and biobanking data (image, clinical, pathological and molecular). The application will include secure management as well.

Everything will be connected to a front-end interface to give to the users a convenient way to manage, submit executions and compare results. The project results might also impact the management of other malignant solid tumours as most of the methodologies proposed in all components can be transferred to other cancer types.

### 1.3.3. In-silico scale models

The objective of the multiscale predictive model is to simulate the neuroblastoma growth as a function of the disease progression and of the chemotherapy treatment of choice, in order to determine the best drug regime based on its forecasted effects in an *in-silico* tumour.

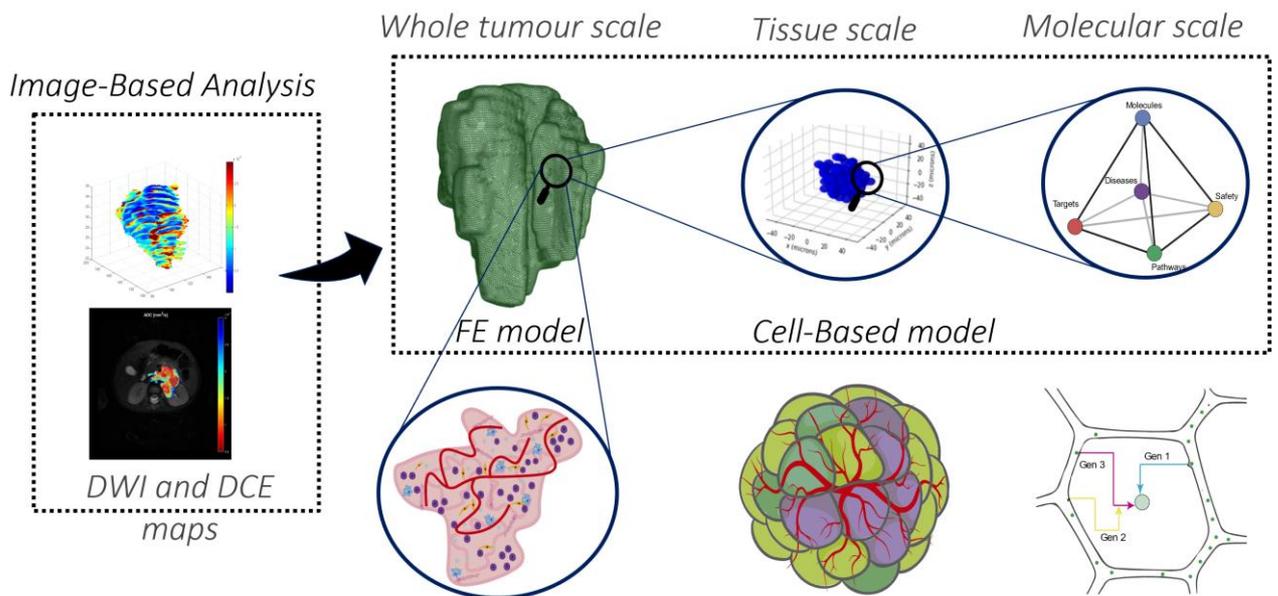


FIGURE 1.8: Scale separations and their models in PRIMAGE project.

The multiscale tumour model is composed of four models in three different scales (Fig. 1.8). The largest scale represents the tumour, and it has two models to simulate two physiological processes of interest: oxygen transportation and mechanical quantities. The oxygen transport model describes the diffusion process of oxygen, and the structural model analyses the mechanical quantities such as stress and strain as a consequence of the growth and the tissue interactions. Both models use ANSYS software and generate as output the evolution of the tumour volume.

The middle scale represents the tissue scale model that simulates the cell interactions in the domain of a single element of the mesh by agent-based modelling (ABM). The goal is to simulate the signalling behaviour of cells together with the genetic influences. It is a hybrid model that represents both neuroblastoma and Schwann cells as discrete agents in a tumour microenvironment represented as a continuous spatiotemporal field. The continuous automation uses a grid of cells to represent space and afford the model an infinite number of states. The cells move between voxels in the grid based on their equation of motion, which is calculated using a locomotion force that is determined stochastically, which makes the tissue model non-deterministic.

The molecular model aims to incorporate the effects of medicine in the multiscale model. The chemotherapy drugs present mechanisms of action that affect subcellular structures not represented at the tissue scale and neuroblastoma treatments vary based on multiple factors like age, the disease stage, tumour histology and MYCN amplification. The molecular model incorporates those factors to return the effects of the drug regime, that is, the inhibition of proteins in the tissue scale domain.

#### 1.3.4. Pre-processing scripts

A piece of the initial data of PRIMAGE orchestrations will be derived from medical imaging and clinical databases.

UNIZAR developed an automatic workflow to reconstruct the patient tumour and generated its finite element mesh. This pre-processing script takes the geometry and image files from the imaging database to generate the tumour mesh and the values of cellularity, oxygen concentration and vascularization of each element using ANSYS and python libraries to do this. The estimation of physiological features from imaging data is done based on works present in the literature.

Regarding the chemotherapy selection, CHMT is developing a pre-processing script to determine a drug regime to be applied to the *in-silico* tumour based on genetic information. However, for a group of cases, there are no data about all the genes necessary to develop a bioinformatic model to determine the personalized effects of chemotherapy. Thus, for these cases, a second pre-processing model is used to combine the genetic data available with other clinically relevant data to determine the optimal drug regime.

As described, simulating a solid growth tumour requires modelling across a wide range of spatial and temporal scales. A corresponding mathematical problem must include the cellular and subcellular dynamics as well as the solid tumour mechanics. The next chapter will describe the proposed scale-separation approach for a patient-specific neuroblastoma growth model.

## 2. The PRIMAGE multiscale model

### 2.1. Scale separation analysis

The scale separation analysis for the PRIMAGE neuroblastoma model was conducted by Barbara de Melo Quintela and is described in detail in (de Melo Quintela et al., 2022). As such it is not part of this thesis. However, as the whole thesis is built on the results of that work, in this section we summarize its key elements.

#### 2.1.1. The idealisation of tumour growth: a mathematical model

For designing a tumour growth model, like any multiscale model, the first key decision regards scale separation. Time and space are manifested in nature as continuum scales and the decision on how to partition them must be led by some key aspects such as the phenomenon of interest to model, the resolution of the instrumentation used for informing and validating and, the computational power available. Neuroblastoma tumours comprehend a huge variety of cells.

Considering:

$\dot{S}_j$  = the function of the rate of the chemical specie  $S_j$

$X$  = a point in the tumour

$t$  = time

$N$  = number of cells inside an infinitesimal element neighbouring the point  $X$

$\chi_k^j$  = consumption rate of the chemical specie  $S_j$  by the cell  $k$

$\sigma_k^j$  = supply rate of the chemical species  $S_j$  by cell  $k$

$I_k$  = type of cell  $k$

$\alpha_k$  = level of differentiation of cell  $k$

$\gamma_k$  = internal state of cell  $k$

$\tau_k$  = telomerase state of cell  $k$

Assuming that a cell can be originated from the neural crest or from the radial glial cells and have two states: replication and death, the probability of a cell  $k$ , inside an untreated tumour, change its internal state ( $\gamma_k$ ) is a function of the cell's type ( $I_k$ ), the differentiation level ( $\alpha_k$ ) and the concentration of each of the chemical species of interest,  $S_j$  where  $j$  is the index of the chemical molecule represent (Eq. 1).

$$\text{Equation 1: } \pi_{\gamma_k}(k(X), t) = \pi_{\gamma_k}(I_k, \alpha_k, \tau_k, S_1, \dots, S_J, t)$$

The probability that a cell proliferates or dies depends on its internal state and the chemical signals of the specific chemical species  $S_j$  encoded by the concentration dynamics in a region (Eq. 2).  $S_j$  are provided across the volume and consumed by the cells.

$$\text{Equation 2: } \dot{S}_j(X, t) = \sum_k^{N \in dV_X} \chi_k^j(I_k, \alpha_k, \gamma_k, \tau_k, t) + \sum_k^{N \in dV_X} \sigma_k^j(I_k, \alpha_k, \gamma_k, \tau_k, t)$$

The rate of changing the number of cells of type I in a volume  $dV_X$  is a function of the proliferation rate of cells of type I, the differentiation rate, and the rate of cell death. The proliferation rate of type I cells inside an infinitesimal volume in a point X at time t ( $r_i dV$ ) is the derivate in the time of its concentration function that depends on all chemical species  $S_j$  in this volume. Thus, by imposing the conservation of mass, it is possible to obtain the  $r_i dV$  function.

$$\text{Equation 3: } \frac{\partial C_i^{dV_X}(X, t)}{\partial t} + \nabla \cdot \left( C_i^{dV_X}(X, t) \frac{\partial u(X, t)}{\partial t} \right) = r_i^{dV_X}$$

The total volume of a tumour at an instant t of time is given by the sum of the cellular volume and the ECM (Extra Cellular Matrix volume). Nevertheless, the ECM volume changes in proportion to the cellular volume. Considering every cell has a similar volume, the cellular volume ( $C^V$ ) is proportional to the number of cells in a volume V.

$$\text{Equation 4: } \frac{\partial V}{\partial t} = k^{ia} \left( \frac{\partial C^V}{\partial t} \right) = k^{ia} \left( \frac{\partial C_S^V}{\partial t} + \frac{\partial C_N^V}{\partial t} \right)$$

These equations are referred to a tumour without treatment. Theoretically, a drug regime should revert its growth or at least reduce the progress of the tumour. Thus, the probability of a  $k$  cell changing its internal state in a tumour with treatment being applied is the probability of doing it in an untreated tumour (Eq. 1) multiplied by the probability of the treatment causing these changes,  $k^{\text{treat}}$  (T) where T is the treatment applied.

$$\text{Equation 5: } \pi_{\gamma_k}^*(k(X), T_l, t) = \pi_{\gamma_k}(I_k, \alpha_k, \tau_k, S_1, \dots, S_J, t) \cdot \pi_{\gamma_k}^{\text{treat}}(T_l)$$

Gathering all the equations we obtain a mathematical model which describes the growth of the tumour.

$$\text{Equation 6: } \left\{ \begin{array}{l} \pi_{\gamma_k}^*(k(X), T_l, t) = \pi_{\gamma_k}(I_k, \alpha_k, \tau_k, S_1, \dots, S_J, t) \cdot \pi_{\gamma_k}^{\text{treat}}(T_l) \\ r_i^{dV_X}(X, t) = \frac{dC_i^{dV_X}(X, S_1, \dots, S_J, t)}{dt} \\ \dot{S}_j(X, t) = \sum_{k \in dV_X}^{N \in dV_X} \chi_k^j(I_k, \alpha_k, \gamma_k, \tau_k, t) + \sum_k^{id} \sigma_k^j(I_k, \alpha_k, \gamma_k, \tau_k, t) \\ r_i^{dV_X}(X, t) = f_p^{i,a}(X, t) - f_d^{i,a}(X, t) = \frac{dC_i^{dV_X}(X, t)}{dt} \\ \frac{\partial C_i^{dV_X}(X, t)}{\partial t} + \nabla \cdot \left( C_i^{dV_X}(X, t) \frac{\partial u(X, t)}{\partial t} \right) = r_i^{dV_X} \\ \frac{\partial V}{\partial t} = k^{ia} \left( \frac{\partial C^V}{\partial t} \right) = k^{ia} \left( \frac{\partial C_S^V}{\partial t} + \frac{\partial C_N^V}{\partial t} \right) \end{array} \right.$$

### 2.1.2. The problem of scale separation

For designing a scale separation, we must provide a justification for each scale separation based on the resolution of the experimental methods and the computational limitations. The growth of a solid tumour was modelled across three scales: tumour scale, tissue scale and molecular scale.

Patient-specific images are used to obtain the geometry and the corresponding biomarkers. A Finite Element Model (FEM) is used to solve the whole tumour and simulate the spatial-temporal diffusion of oxygen and nutrients as well as the vascularization. Using particularization, this information is passed to the tissue scale model, which will predict how the cells evolve based on data obtained from the tissue model regulating differentiation, proliferation, hypoxia, and matrix

formation. An Agent-Based Model (ABM) describes these phenomena from cell-to-cell and cell-to-matrix interactions. Each agent represents a cell, which has its state updated reflecting behaviours like proliferation and apoptosis/necrosis using also genetic and molecular factors and the conditions of the microenvironment (oxygen concentration, chemotherapy, vascularization, cell crowding etc).

The volume variation of the Reference Volume Element (RVE) explicated in the ABM is incompatible with other RVE inside the tumour. The RVE is simulated inside the ABM unrestrictedly, that is, the volume changes without influences of the nearby RVEs. So, if an RVE shrinks, some gaps emerge in the mesh as well as if an RVE grows, all RVEs must to be dislocated and rearranged to obtain the new mesh shape. Then, through an interactive process, we ensure compatibility between all RVEs and connect them, by means of a homogenization technique, from the tissue scale to the tumour scale. The macroscopic scale mechanical deformations and a new diffusion of nutrients is simulated before calling the particularization to simulate the tissue scale again.

The cell model can consider the effect of some treatment, between the tumour and tissue scale to update the effect of the chemotherapy on the cellular pathway inside a microenvironment at the tissue scale. It is a probabilistic model that does simulates the molecular phenomena. The model only calculates how the drug regime will impact the cell state changes based on clinical and genetic data and the vascularization status.

### 2.1.3. *Scale separation analysis for the Neuroblastoma model*

#### 2.1.3.1. Scale and dimensional analysis for Neuroblastoma

Neuroblastoma is a strongly heterogeneous cancer, with a large variety of different clinical outcomes. Its characteristics are shared with several other types of cancer what makes neuroblastoma usable as a paradigm of cancer diseases and a great quick off point to the development of models that are applicable to other cancers.

In order to model any phenomenon, we need to identify the spatiotemporal scales where the phenomenon of interest occur. A scale is described in terms of grain and extent. The grain is the difference between the temporal/spatial resolution and the fastest/smallest characteristic to be observed. The extent is the opposite of the grain, that is, the smallest values between the upper limit of the temporal/spatial resolution and the size of the slowest/largest phenomenon of interest.

A neuroblastoma can be as large as  $800 \text{ cm}^3$  (around 10 mm of diameter) or as small as  $1 \text{ cm}^3$ . The drug regime effects on the tumour must be modelled at the single-cell scale. Thus, the extent is equal to 10 mm while the grain should be tiny as a molecule's size which is in order of  $5e^{-10}$  nanometres.

The temporal scale range is more complex. Neuroblastoma cases are typically treated with up to three chemotherapy cycles about every eight weeks. So, for a model that aims to simulate the whole chemotherapy's duration, the temporal extent must be set as 24 weeks ( $10^7$  seconds). While the grain is defined by the chemical reactions. Here, it is very hard to be specific, but they usually happen in an order of  $10^{-2}$  seconds. Thus, we can conclude that a full resolution model needs a spatial extent up to  $10^{-1}$  m and a grain of  $10^{-8}$  m and a temporal range from  $10^{-2}$  until 107 seconds, that is, we need a span of 9 orders of magnitude for both, temporal and spatial ranges. This is too large to be simulated in a continuous resolution so a multiscale approach must be applied.

#### 2.1.3.2. The limits of scale separation

All limits of space and time are determined by the problem itself. For the tumour model, the spatial extent is set by the size of the solid tumour and the grain is set by the definition of the image medical instrumentation used to obtain the 3D tumour geometry and the number degrees of freedom that the finite element method uses. Regarding the temporal extent, it is defined by the duration of the treatment (up to 24 weeks) while the experimental temporal grain is the minimum distance between two successive imaging verification. If we assume this will be done one for each 1 – 3 months the evolution of the tumour provided by the model should smaller than this time range, so at least 2 weeks.

About the tissue model scale, it is convenient to set the spatial and temporal extents as the same as size of the respective grain of the tumour scale. The spatial grain of the tissue model should be as small as the smallest component of interest, in this case, the cells. However, there is also a computational limitation by the amount of agents the model can manage at the same time. The temporal extent is restricted computationally by the maximum timestep that guarantees a tolerable discretization error for the tissue model (Fig. 2.1)

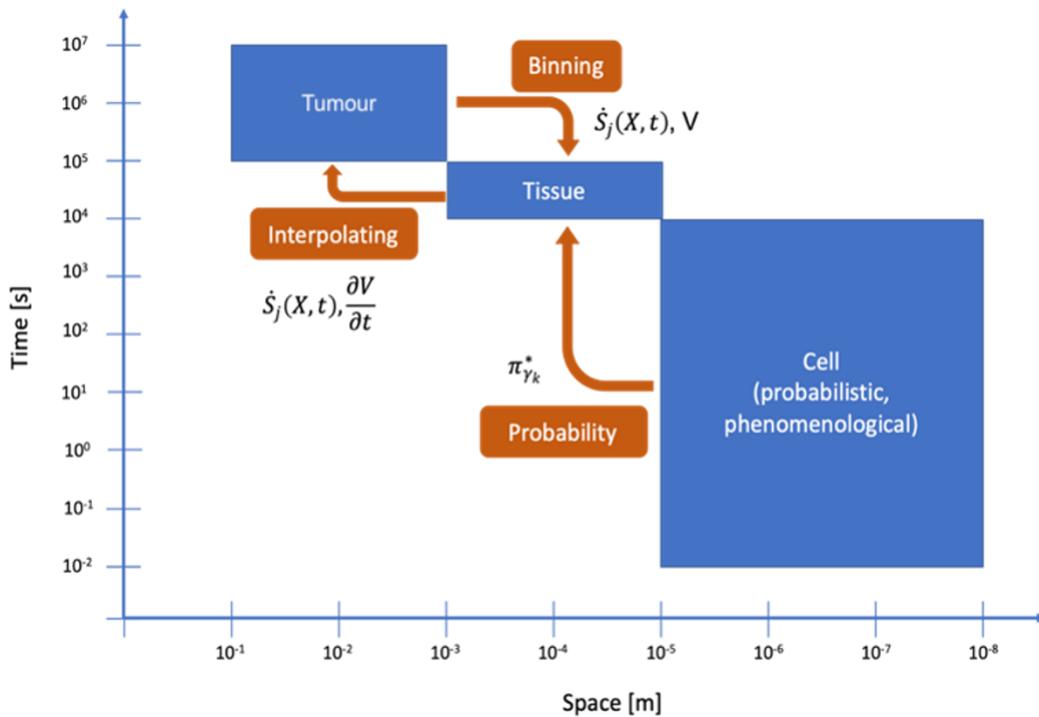


FIGURE 2.1: Scale separation for a multiscale model of tumour growth. Reprinted with permission from (de Melo Quintela et al., 2022).

The theoretical analysis conducted in (de Melo Quintela et al., 2022) was conducted at the beginning of the PRIMAGE project, before the multiscale had been fully developed. The final implementation of the molecular-scale model differs to some extents from what was assumed in the scale separation analysis. The probabilistic and phenomenological nature of the molecular-scale model does not imply the implementation of molecules mechanisms once the model only returns the probabilities forecasted for the relevant pathways for each voxel, so this model does not have a strict grain. Also, the cell model is not coupled to the tissue model, the probabilities provided affect only the rates of cellular events in the tissue model. The model runs multiple times for updating the probabilities after the remeshing and because its outputs depend on the vascularization level given by the tumour model.

## 2.2. The orchestration target: the PRIMAGE model

The orchestration is proposed as a dataflow strategy, whereas from the orchestrator point-of-view each single model is identified like a black box that gets a set of input files to produce a set of output data. The orchestrator manages all the data during a simulation and register them in an internal database, to keep track of the full execution. Tumour growth models generally have four sets of inputs: Imaging data, clinical data, genetic data, and the treatment choice (treatment and treatment duration), and returns as output the updated tumour volume represented by the variation of the tumour volume, its cellularity, its oxygen concentration and its shape (Fig. 2.2).

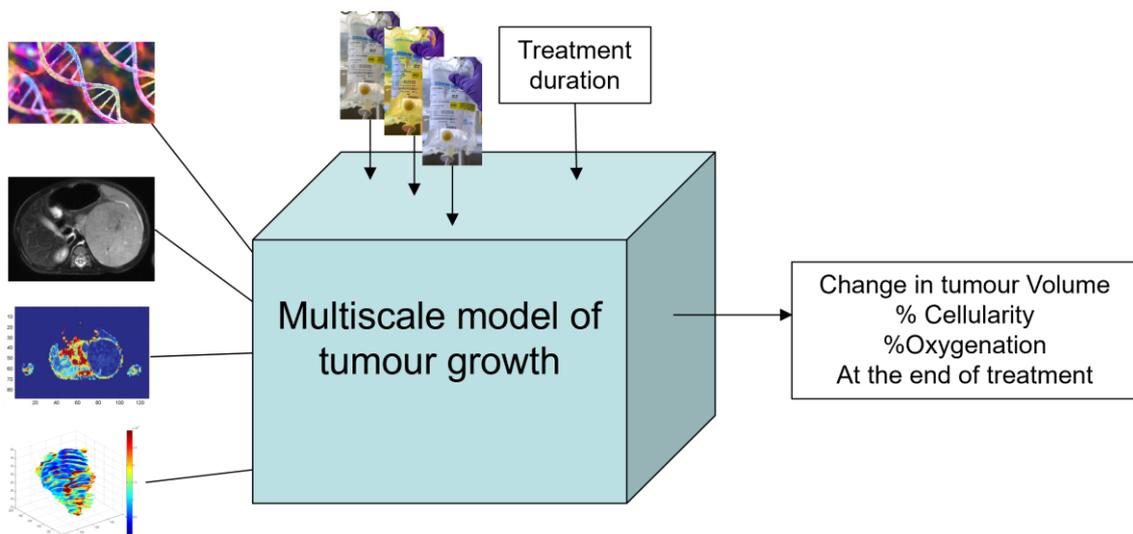


FIGURE 2.2: General template of tumour growth models.

### 2.2.1. The inputs

The PRIMAGE orchestration starts by downloading a set of imaging data, genetic data, and clinical data of a patients, which are used at different scales of the neuroblastoma model.

Imaging data represents the biggest files of this application, they come from GIBI<sup>1</sup> database, a source of imaging biobank of clinical patients, pseudonymously linked to each patient and used by a pre-processing model, developed by UNIZAR, that generates a mesh of the tumour shape and the set of tumour data to start the first model, here called of oxygen transportation model. These files give for each element of the mesh its oxygen concentration, initial cellularity of the neuroblasts alive cells and the vascularization. GIBI also provides histology data to be used by the ABM at the tissue scale (Fig. 2.3) The Appendix 1 provides further details of the I/O of all models.

The genetic data is obtained from the QUIBIM<sup>2</sup> database. QUIBIM is a biotech company that work on the transformation of imaging data into actionable predictions to improve patients' outcomes. The imaging data is accessed in a data lake through OneData services<sup>3</sup> and the genetic data is accessed using a QUIBIM API. The patients are linked in both databases by pseudoanonymisation which is used by the PRIMAGE platform to register the multiscale model simulations.

<sup>1</sup> [www.hospital-lafe.com](http://www.hospital-lafe.com)

<sup>2</sup> [www.quibim.com](http://www.quibim.com)

<sup>3</sup> [www.onedata.it](http://www.onedata.it)

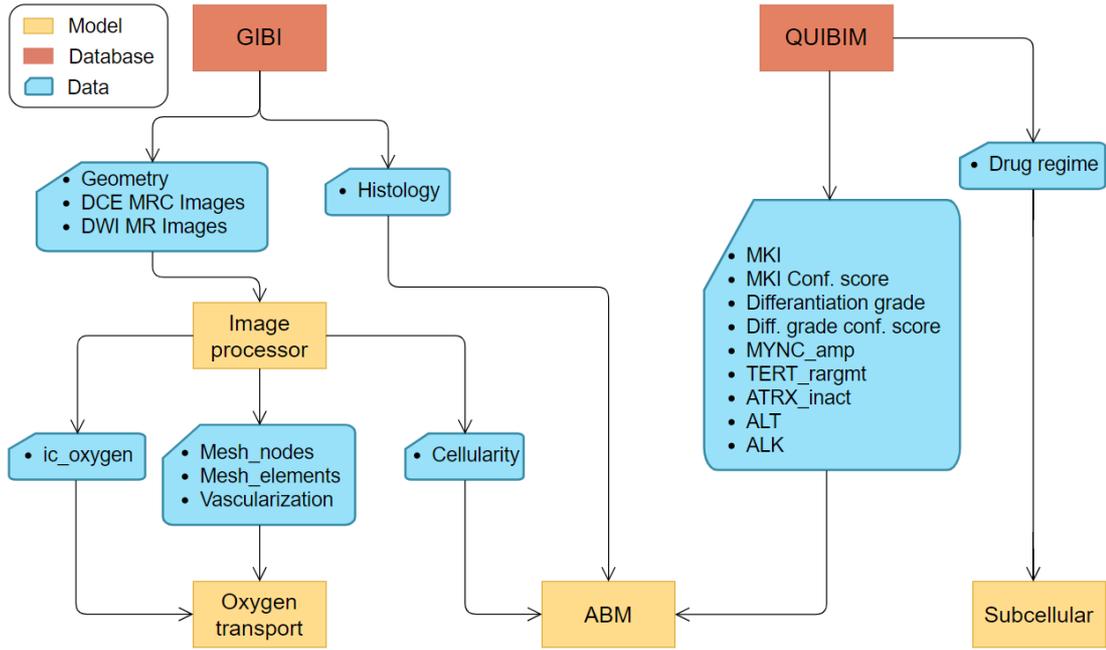


FIGURE2.3: Input data for PRIMAGE solid growth tumour model.

### 2.2.2. The component models

Four models compose the PRIMAGE orchestration: oxygen transport (UNIZAR), chemotherapy model (CHMT), Agent-Based Model (USFD), and structural model (UNIZAR). Two models work at the tumour scale (oxygen transport and structural models), one in the tissue scale (ABM) and one in the cellular scale (chemotherapy model). For starting a simulation, the image processor (UNIZAR) does a pre-processing task, working on images gotten from GIBI image database and returning a mesh of a tumour shape and the initial values to the oxygen transport model and the Agent-Based Model (ABM).

The oxygen transport model is a Finite Element Model (FEM) implemented on ANSYS software to compute the diffusion of the oxygen through the whole tumour geometry. This model receives the geometry files, the oxygen concentration and vascularization ( $K_{trans}$  value) of each element as input and returns the updated oxygen concentration and the volume for each element of the mesh (Fig. 2.4). In the whole orchestration the oxygen concentration is computed as the dimensional scaled by 72mmHg while the volume is in cubic millimetres for the tumour scale models.

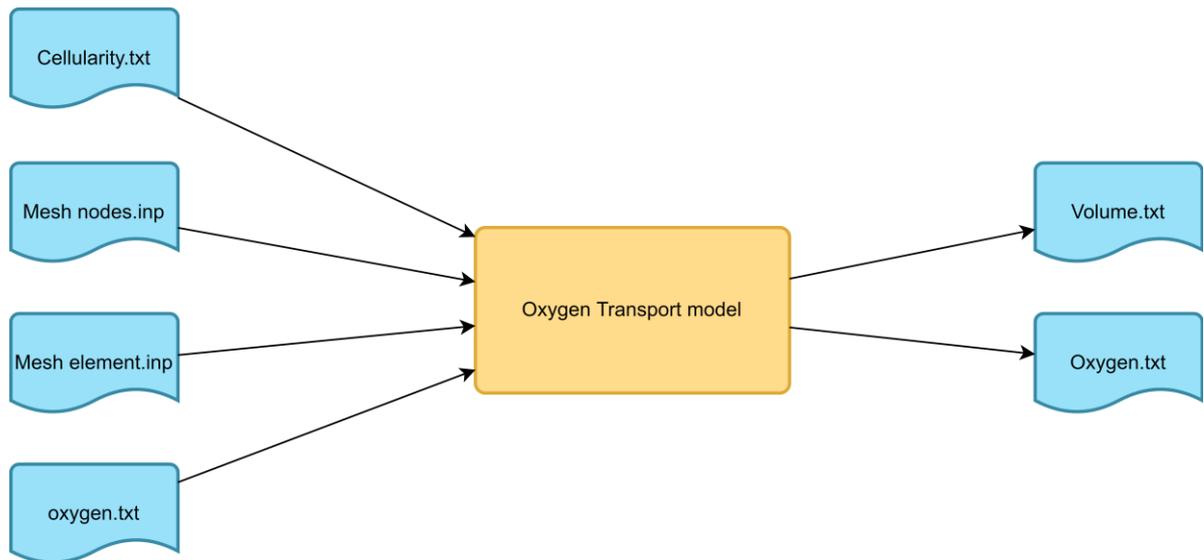


FIGURE 2.4: Oxygen transport model I/O files

The next step is the selection of a sample of elements in a process called by particularization in order to run the chemotherapy model (subcellular model) and the ABM.

The subcellular model is a Python implementation to estimate, for the sample of elements, the drug inhibition probability on the proteins based for the drug regime previously defined. The model works on the subarray of elements selected on the basis of their vascularization value to return a subarray of outputs that will be consumed by the instances of the tissue model (ABM) (Fig. 2.5).

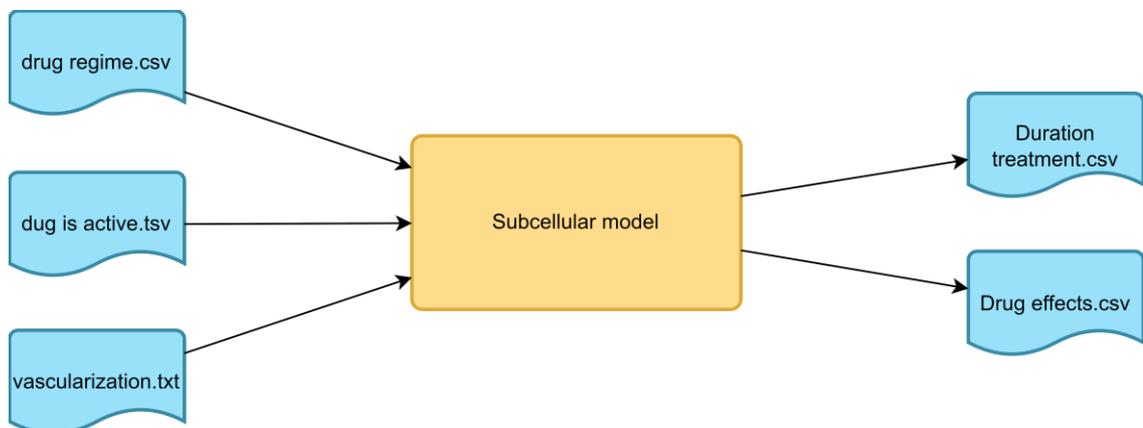


FIGURE 2.5: Subcellular model I/O files

The ABM simulates the multicellular phenomenon, calculating the interactions between neuroblastic and Schwann cells in a patch of the tumour (element). This model is run once for each element sampled. The model receives information of the patch (e.g., volume, oxygen concentration, cellularity etc.) and static data from GIBI (image DB) and QUIBIM (Genomics DB) databases, like histology, differentiation grade, and some mutations data. At the first loop execution, the model determines a piece of variables (e.g., telomer length, death signals etc) by the histology data for, in the following loops, to receive these data updated as inputs (Fig. 2.6).

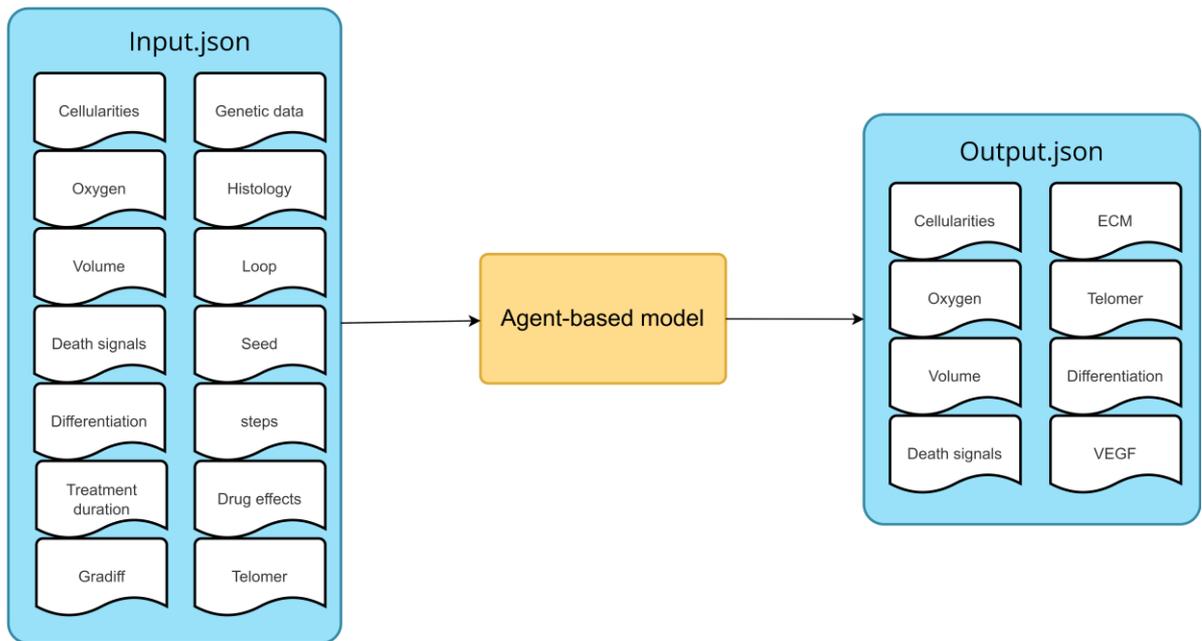


FIGURE 2.6: ABM I/O files

After all the ABM executions finish, the homogenization process begins. This process will bring the orchestration back to the whole tumour scale, using the values obtained for the sample of elements to estimate all the values for the other elements. The homogenized data will be used to run some scripts that will generate the input files for the structural model and some input files for restarting the cycle (if this is the case).

Given those inputs, the structural model, which is also a FEM, will run for the whole mesh using the volume variation and the materials properties to calculate the new geometry and volumes (Fig. 2.7).

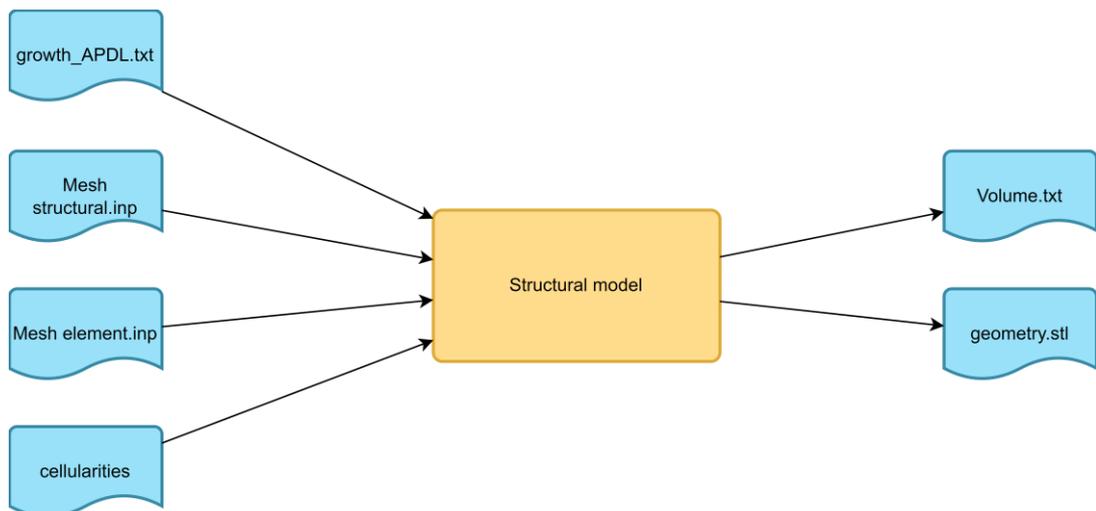


FIGURE 2.7: Structural model I/O files

At this point, there is a verification to check if the time of simulation of the orchestration was reached or not. In a positive case, the structural model outputs compose the result of the multiscale model. Otherwise, a new cycle starts, doing a remeshing and changing the non-static initial inputs for

inputs generated by the outputs of the models. The whole dataflow of PRIMAGE multiscale model is summarized in the figure 2.8.

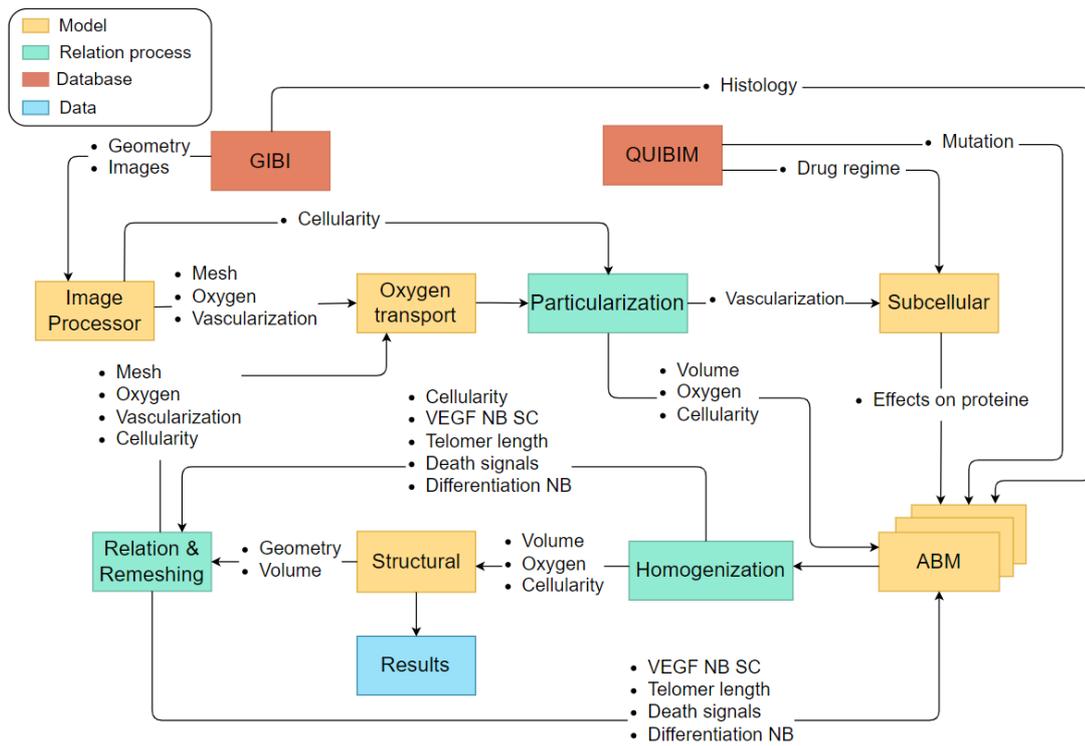


FIGURE 2.8: Topology of the PRIMAGE multiscale model.

### 2.2.3. The outputs

The outputs are composed of the meshes files representing the new tumour's geometry after the phenomenon simulated, together with its oxygen concentration, cellularity, and the updated vascularization. The orchestration is cyclic, that is, after the last model run, the first model runs anew, and this loop is repeated until the whole treatment time is simulated. Every loop output its results to the database, which makes possible to follow the variations of tumour's size in a certain time span.

## 3. The orchestration framework: VPH-HF v3

### 3.1. Introduction

#### 3.1.1. *The need for models' orchestration*

The orchestration of predictive models is, in general complex but can be simplified using an orchestration software framework. Such a framework would have to ensure centralised management of the *control flow* (the set of executions instructions to be imparted to each model during the orchestration) and of the *data flow* (the necessary handling and transformation of data between the output of a model and the input of another).

The framework also includes the *relation models*, which are scripts to pick outputs of one or more models together with other data and transform them into inputs to the following model.

#### 3.1.2. *A critical review of existing orchestration frameworks*

Several phenomena in engineering and science are manageable through multiscale modelling. Various software libraries were released to support the construction and orchestration of multiscale models. The software aims to facilitate the coupling of the models, enable access to HPC systems and automatise the management of multiscale simulations. Several frameworks have been previously proposed to address this sort of problems such as Taverna (Wolstencroft et al., 2013), MUSCLE2 (Borgdorff et al., 2014, p. 2), FABSIM (Groen et al., 2016), AiiDA (Pizzi et al., 2016) and a lot more.

Multiscale modelling is a divide-to-conquer approach where single-scale models are combined to form a multiscale model. This scale separation map is often encoded in the Multiscale Modelling Language standards (MML) (Falcone et al., 2010), and years later, a work (Alwayyed et al., 2017) released the Multiscale Computing Patterns (MCP) that has introduced abstractions to more generic multiscale software.

MUSCLE introduced an approach of component-based architecture, defining a multiscale model as a set of coupled single-scale models. The framework was designed for complex automata modelling and multi-agent computing. Later, MUSCLE2 (Borgdorff et al., 2014, p. 2) was released, tailored for distributed multiscale computation, including MML, allowing a clearer formalisation of complex coupling of models. On MUSCLE 2, the model, called sub-models, can run independently, and a connection can be established between two sub-models. The software focused on cyclically coupled models with all sub-models actively sending feedback during the entire simulation and supports multilanguage component models.

Meanwhile, Taverna was released (Wolstencroft et al., 2013). Taverna was a workflow tool to join distributed web services and local tools into complex analysis pipelines. The software could run these pipelines locally or in more extensive computational infrastructures through the Taverna server. Using Taverna, there was no necessity to install tools and data sources locally once a significant part of the computational processing was done remotely. In 2014 the project moved to the Apache incubator, and in 2020 was definitely closed.

The Hierarchical Multiscale Simulation (HMS) framework (Knap et al., 2016) incorporates stand-alone executables to be written in any programming language in the multiscale model. It implements a runtime system to schedule and execute the models. In 2018, the framework was extended to address an adaptative surrogate method to reduce the computational cost (Leiter et al., 2018).

In the context of the EU project VPHOP, the team of Prof Viceconti developed one of the first of these frameworks, called the Virtual Physiological Human Hypermodelling Framework (VPH-HFv1)

(Viceconti, 2012). VPH-HF v1 was based on a bus architecture for the communication between models and required each model to be wrapped with a C++ code that abstracted its control flow. VPH-HF v1 allowed very complex control flows but offered little support for the management of the data flow. During the CHIC project, funded by the FP7 EU framework, the Insigneo Institute developed a second version of this framework, VPH-HFv2 (Viceconti et al., 2018). VPH-HFv2 was developed with an architecture inspired by the concept of modularity that permits each component to be used in isolation. The framework wrapped both Taverna and Muscle2 as workflow management libraries. Although these approaches were designed to be flexible and customisable, they became highly complex to maintain and re-reuse mainly because the architectures were focused on control flow rather than data flow.

Since multiscale models often represent a complex workflow of complex, large-scale parallel models, there is a demand for deploying the models in multiple high-performance computers with very different systems. Fabsim (Groen et al., 2016) is one example of a tool used to overcome this situation. Another is the AiiDA library (Pizzi et al., 2016) (Automated Interactive Infrastructure and Database). AiiDA is "an open-source Python infrastructure to help researchers with automating, managing, persisting, sharing and reproducing the complex workflows associated with modern computational science and all associated data"<sup>4</sup>. Implemented in Python, AiiDA has been designed as an intermediate layer between the user and the HPC resources.

Groen *et al.* did a recent review of the multiscale computing software and analysed the value of several tools regarding the support in the development of multiscale models (Groen et al., 2019). It was observed that most of the available tools focus on the implementation of the workflow and less on instantiation and execution and there is only marginal attention to production and optimization steps.

The present work proposes the development of an updated framework, referred to herein as the VPH-HFv3, as a part of the PRIMAGE project. Such framework, a complete re-writing with respect to the previous versions, aims to orchestrate several models, which are currently under development, using a simple architecture, inspired on the High-Level Architecture for distributed simulation, easy to maintain and with high reusability; the focus is on the effective management of the data flow, which is in most computational biomedicine simulations the most important aspect.

### 3.1.3. *The High-Level Architecture for distributed simulation*

The architecture designed for the VPH-HFv3 is based on the HLA (High-Level Architecture). We used the HLA formalisms and the functional design phase as a template to elaborate the framework architecture aiming to take advantage of the HLA conveniences. The High-Level Architecture (HLA) (Dahmann et al., 1998) gives the specification of a common technical architecture for use across all classes of simulations in the US department of defence. It was developed with the goal of defining a common simulation infrastructure to support interoperability and reuse in different simulations.

The HLA was created assuming there is no single simulation that can satisfy all requirements for all types of users and uses. The intention was to provide a structure to support the reuse of capabilities in different simulations reducing, then, the cost and time needed to create a synthetic environment for a new application. In fact, such architecture has large applicability including areas such as education and training, engineering and even entertainment.

The first key components of HLA are the federates, or more specifically the simulations themselves. A federate can also be a manned simulation, a supporting utility, or an interface to a live player on the instrumented facility (Fig. 3.1). The only constraint HLA imposes regarding a federate is that it must include specific features to allow the objects in the simulation to interact with other objects exchanging data supported by the RTI (RunTime Infrastructure).

---

<sup>4</sup> <https://www.aiida.net/>

The second key component is the RTI. It provides a set of services to enable federate-to-federate interactions, as well as federation management and functions of supporting. The third and last functional component is the interface to the RTI. Which provides the interaction federate-RTI, to access the RTI services and to respond to requests from the RTI.

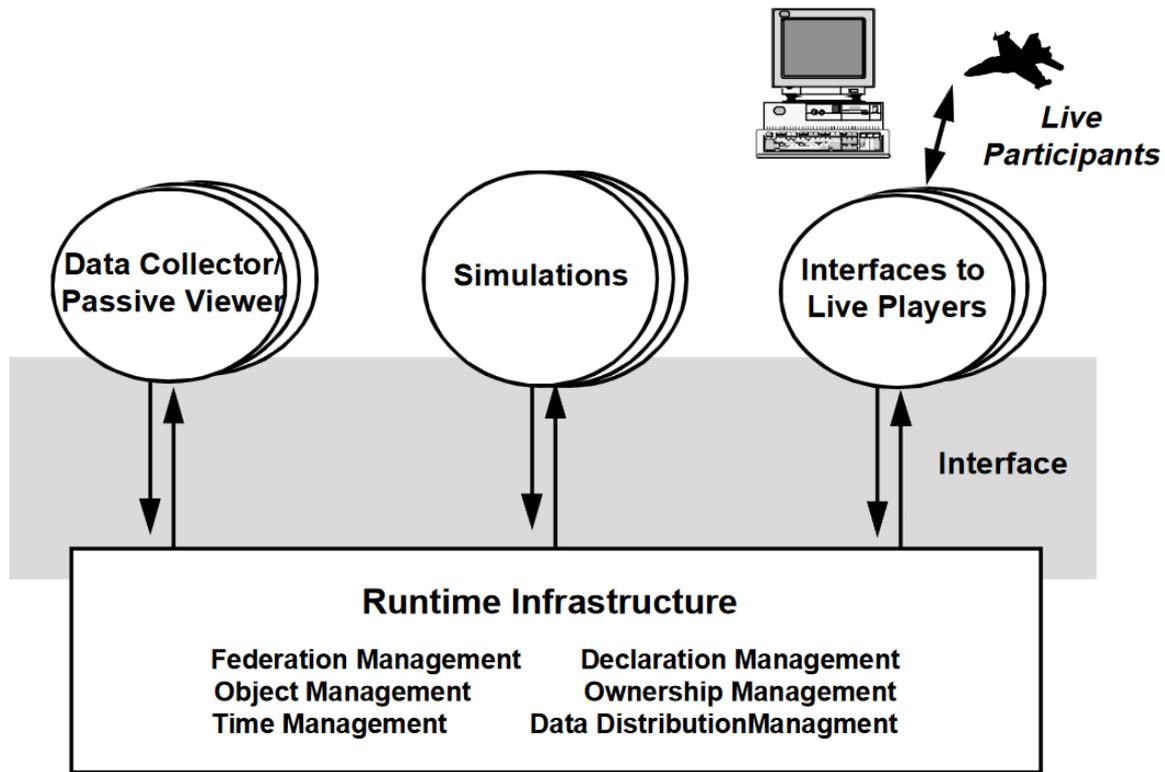


FIGURE 3.1: Components of the HLA.

The HLA consists of three parts:

1. Rules
2. Object Model Template (OMT)
3. Interface Specification (IFSpec)

The IFSpec describes the runtime services provided by the RTI to the federates and vice-versa. There are six types of IFSpec as described in table 3.1. The Object Models are a description of the essential elements of a simulation (or federation). Two types of Object Models are defined in the architecture: the Federation Object Model (FOM) and the Simulation Object Model (SOM). They describe the set of objects, attributes and interactions that are shared across a federation and the simulation (federate) in terms of the types of objects, attributes, and interactions respectively.

CATEGORY	FUNCTIONALITY
Declaration Management	Establish intent to publish and subscribe to object attributes and interactions
Object Management	Create and delete object instances Control attribute and interaction publication

	Create and delete object reflections
Ownership Management	Transfer ownership of object attributes
Time Management	Coordinate the advance of logical time and its relationship to real time
Data Distribution Management	Supports efficient routing of data
Federation Management	Create and delete federation executions Join and resign federation executions Control checkpoint, pause, resume, restart.

Table 3.1: Interface Specification types in High-Level Architecture.

Then we have the HLA rules which are divided into two groups: Federation and federative rules. The federation rules determine:

- The federation must have a FOM documented in accordance with the OMT.
- All simulation object instances shall be in the federates, not in the RTI.
- During a federated execution, all exchanges of FOM data should occur through the RTI.
- During a federated execution, joined federates should interact with the RTI in accordance with the Interface Specification.
- During a federated execution, an instance attribute must be owned by at maximum one federate at a given time.

While the federate rules say that federates shall:

- Have a SOM in accordance with the Object Model Template (OMT).
- Be able to update and reflect any instance attributes and send and receive interactions in accordance with the specifications in their SOMs.
- Be able to transfer and accept ownership of instance attributes dynamically, in accordance with the specifications in their SOMs.
- Be able to vary the conditions (e.g., thresholds) under which they provide updates of instance attributes, in accordance with the specifications in their SOMs.
- Be able to manage local time in a way that will allow them to coordinate data exchange with other members of a federation.

The rules define the main principles for the whole architecture, the interface specification defines a set of runtime services to support distributed simulations while the object model template ensures the object model interactions. The HLA components given by the rules, the interface specification, and the object model template provide a general essential tool for interoperability and reuse which can be adapted for a huge variety of problems.

### 3.2. The VPH-HF v3 architecture

The main goal of the work is to develop a framework to orchestrate multiscale models using the PRIMAGE project as a reference for the development of the tool. The architecture was developed based on HLA specifications aiming to create a framework adaptable to any problem and with a high level of independence between the parts in order to be easier to maintain and to release it as an open-source tool.

### 3.2.1. Overview

In the PRIMAGE project, we decided to undertake a complete re-writing of the VPH-HF, with two important specifications in mind: the first is that the field of In Silico Medicine is developing several multiscale models that require a very basic control flow management, but a fairly sophisticated data flow management. The second is that in order to be a sustainable Open-Source project, the VPH-HF software must have a simple architecture, easy to maintain and with high reusability. The result, called VPH-HFv3 was developed around a few, simple and clear design choices:

- It is a module-based architecture where all the communication between the modules is done by triggers in the database.
- Aiming to reduce the interdependencies and make the flow less complicated, was defined the rule that each module polls for only one database event and writes in no more than one table on the database (except for the module responsible for registering the orchestration results).
- The data flow is mediated through a relational database, which stores all intermediate results as well as the initial inputs and the results.
- Each model being orchestrated (hereinafter referred to as component model) has a control flow limited to three commands: start, stop, and kill.
- Each model being orchestrated does not require any wrapping, except a serialization script that extracts from the databases the model's input data and prepare them as required, and a de-serialization script that reads the model's outputs and de-serialises them in the database.
- Between the two models, there is the necessity of implementing a connection between them. Such relation is implemented into the orchestrator in a module called relation module.

The resulting conceptual design is depicted in fig. 3.2.

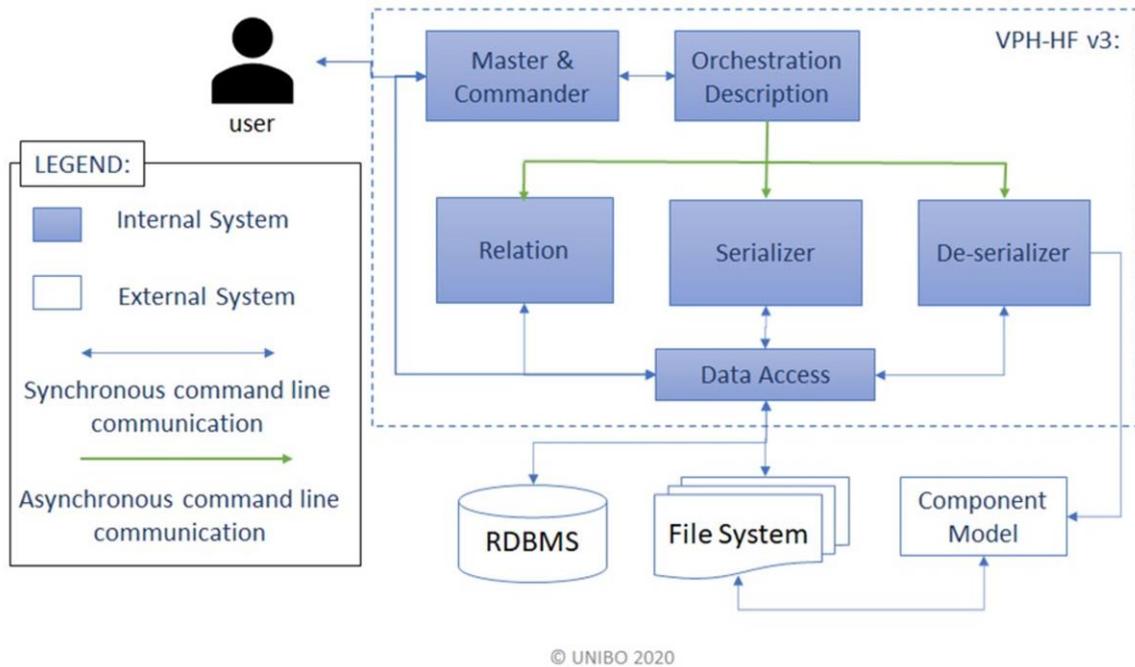


FIGURE 3.2: VPH-HPv3 conceptual design

The architecture is composed of four types of modules that work in parallel in an interdependent way. They are called: Master & Commander, serialisation, relation, and de-serialisation. Each of them, after initialisation, polls the database periodically waiting for a specific condition to do its job and then, to do an operation in the database that will activate the trigger of the following model in the topology workflow. Each instance of the model is responsible for specific tasks and is developed for specific model versions, that means changes in the I/O of a model implies on updates in the modules instantiated for this model. Before getting into details, we need to define the database architecture.

Topology refers to a graph representing the set of rules that defines the control flow and the data flow of a structure, in this case, the multiscale model. There are four types of topologies such as linear or acyclic single-child graph, when each model runs separately and only in one moment at the orchestration; acyclic directed single-child graph, when at some point, at least one model has several simultaneous runs before going further in the simulation. There are the cyclic topologies, when after the last model run, some other model is called again forming a continuous cycling that will run until some criteria be satisfied; and cyclic multi-child topologies, given for the cases where beyond forming cycles, inside them at least one model needs to run several simulations before going to the next model (Fig 3.3). PRIMAGE multiscale model represents a cyclic directed multi-cycle graph.

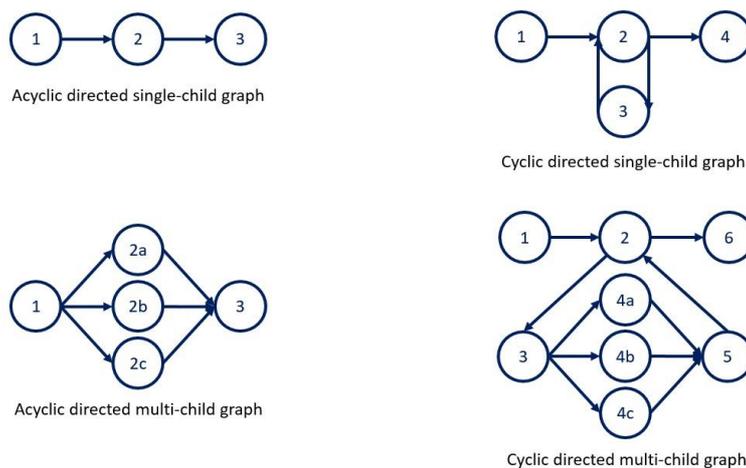


FIGURE 3.3: Topology types

### 3.2.2. Database

The database is a crucial aspect of the framework; it is where all workflow instructions for the orchestrator appear. All the flow is oriented by triggers on the database since each model polls the database waiting for insertion or update which signals the start of its job and, consequently, gives an answer to the next module writing on the database. For example, the Master & Commander module inserts the initial input on the database to start the serializer module of the first model, which when finished will write on the database to satisfy the polling made by the de-serializer module and then go on following the topology rules until the end.

Before going further, we need to define the terminology that will be used hereafter.

- A *component model* (sometimes simply referred to as *model*) is a model with predictive purposes. It is defined by an executable file, its I/O types, and the instructions to run the model.
- A *model type* represents a group of component models that have the same input and output format files and the same call command. These are usually different versions of the same model.
- A *simulation* represents the single execution of a component model. That is, a simulation has its own inputs and outputs although they are not the I/O of the multiscale model.
- An *orchestration* is a set of simulations. It is defined by everything associated with a complete execution of a multiscale model, including all simulations and their ordering, all I/O (intermediate results for an orchestration point-of-view) and all parameters of the execution. So, the I/Os of an orchestration are the I/Os of the multiscale model.
- A *job* represents a call of a model execution in the cluster. It can be associated to one or more simulations.

On the database, each model has a table to store the inputs and another one to store the outputs and the orchestration also has its own input and output tables to store the initial inputs and the final outputs. The default template is shown in figure 3.4. Both tables have their IDs, a field to store all filenames (`names_list`), a pointer to the respective folder (`path`) and the register of when it was created (`datetime`). The input table is linked to the previous simulation by the ‘`provenance_sim`’ attribute or by ‘`initial_input`’, if it represents the first simulation by the ‘`provenance_initial_input`’ foreign key.

The input table also has particular attributes to save the input data, which are different for each model type. All relevant data are registered in the database aiming to have the simulation reproducible regardless the file system.

The output is connected to the input in a relation of one to one represented as 1:n, that is, each input can be associated to 1 or more outputs once for stochastic models, the same input can generate multiple different outputs. To avoid redundancy in the database, only the pointer to the output files is saved in the DB once these data will be written in one or more input tables in the next steps of the workflow.

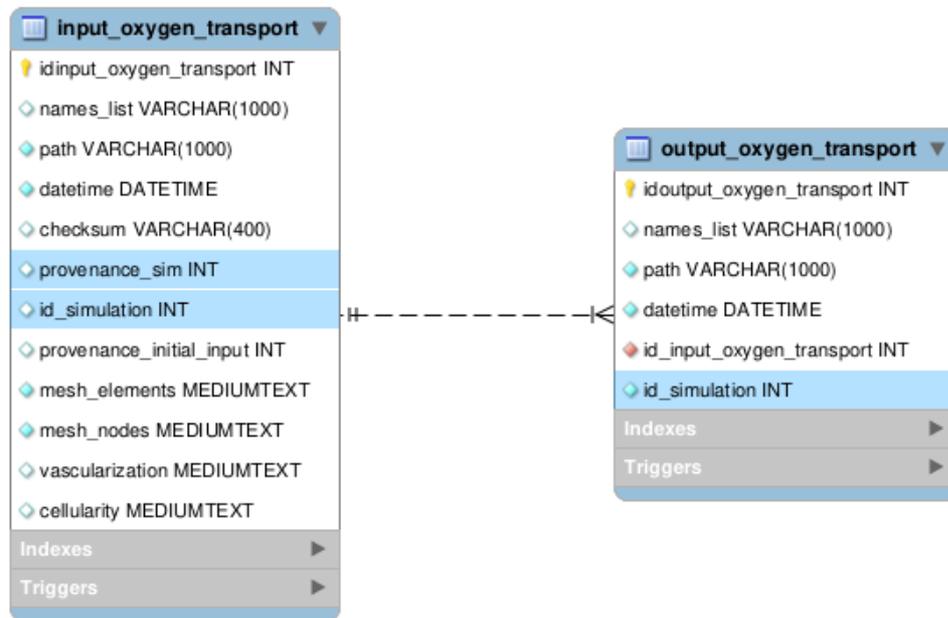


FIGURE 3.4: Example of input and output tables

The database schema has a table to save orchestration data and another table to save simulation data in order to keep the executions tracked. These registers also save the metadata of the simulation such as user, time of start, time of the end, execution time and all simulation data such as number of GPUs used, total treatment time simulated, size of particularisation etc.

Several topologies work in loops, also called *interactions*, so when the last model ends, the first one runs again and so on until the simulation time reaches the total time of the requested multiscale model simulation. Given this, the simulation is associated with a simulation of provenance to not miss the reference of all simulations of the same interaction. A simulation of provenance corresponds to the first simulation of a loop and in order to avoid repetition, it is stored in a table with all parameters necessary for the interactions (Fig. 3.5).

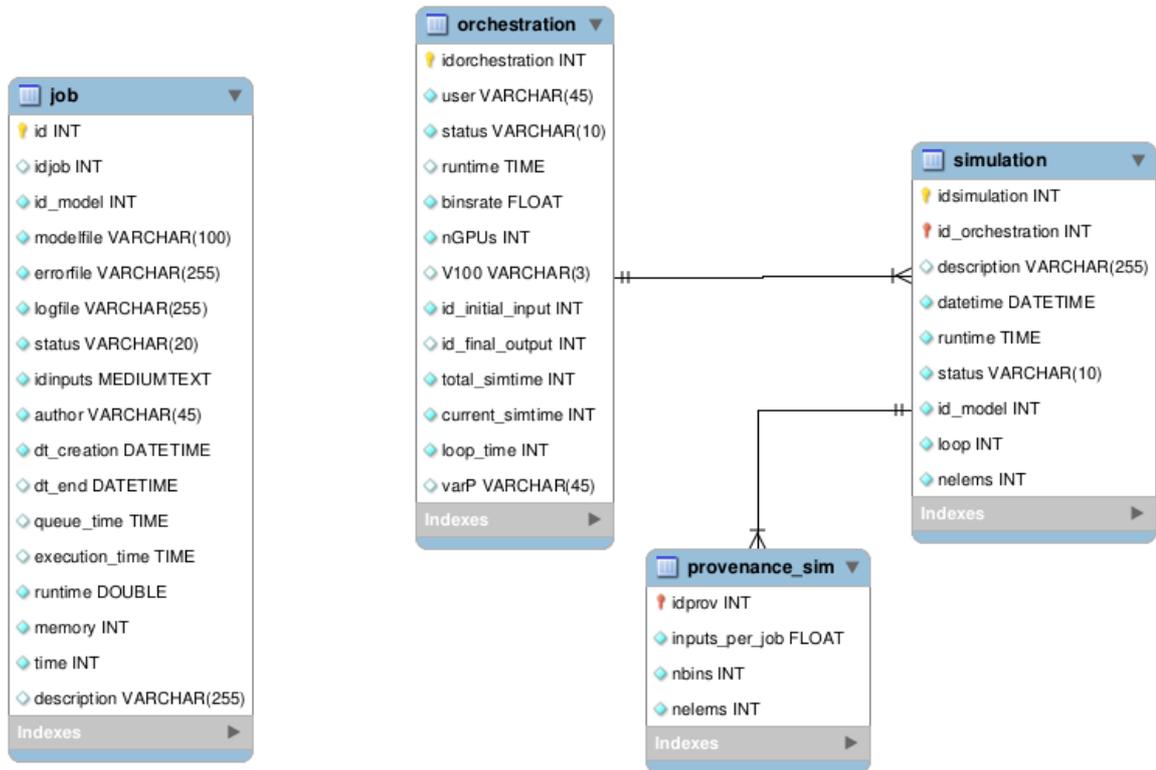


FIGURE 3.5: Job, Orchestration, simulation and ‘provenance\_sim’ tables in the database architecture

The models are registered in the database to ensure that the simulation is reproducible. The model table keeps the model filename, its path, its author, the creation of insertion and a text description. However, as each model has a different set of I/O, the orchestrator needs to be aware of which I/O files are expected. Thus, the column names for each model I/O are being stored in the database. As a set of versions of the same model usually have the same I/O, so, to avoid repetition they are stored separately in a ‘model type’ table which stores also the default time to run this model. This model type ID is also used to pick the correct functions to run the model and prepare the input and output files.

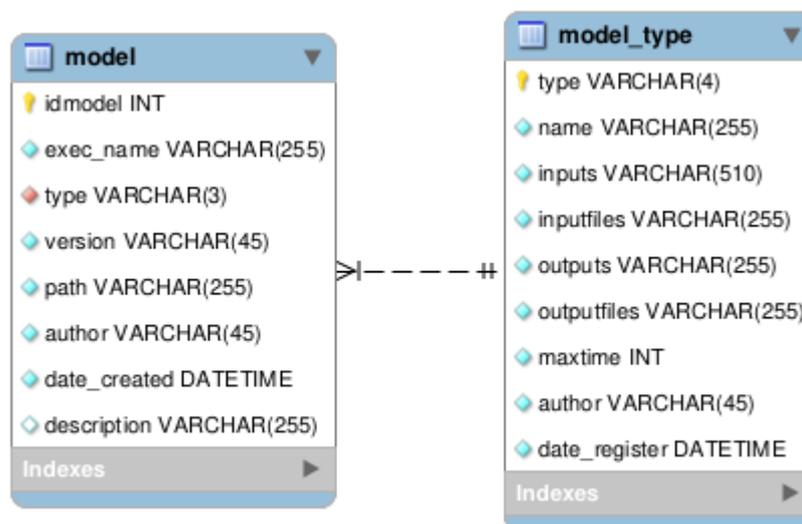
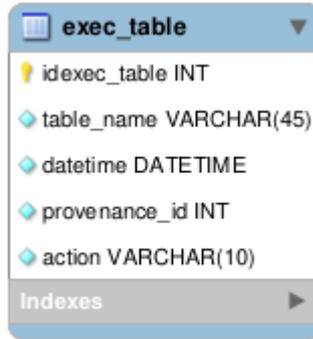


FIGURE 3.6: Models' tables

All the polling is done exclusively in the 'exec\_table'. Each event of relevance has a corresponding trigger to register it in the 'exec\_table'. Doing so, this table assumes the role of a to-do list for the orchestrator. Keeping it in a separate table and using triggers make it possible to continue interrupted simulations when for example the orchestrator is cancelled by the user or because of a timeout. In this way we can also obtain some gain on performance since the 'exec\_table' is empty most of the time, making the polling faster and also ensuring the polling will not get slower once doing the poll in the tables directly would consume more time as long as the tables become.



#	idexec_table	table_name	datetime	provenance_id	action
1	68973	output_ABM	2022-10-18 18:05:14	1	insert
2	68974	output_ABM	2022-10-18 18:05:14	2	insert
3	68975	output_ABM	2022-10-18 18:05:14	3	insert
4	68976	output_ABM	2022-10-18 18:05:14	4	insert
5	68977	output_ABM	2022-10-18 18:05:15	5	insert
6	68978	output_ABM	2022-10-18 18:05:15	6	insert
7	68979	output_ABM	2022-10-18 18:05:15	7	insert
8	68980	output_ABM	2022-10-18 18:05:15	8	insert
9	68981	output_ABM	2022-10-18 18:05:16	9	insert

FIGURE 3.7: Polling table in the database architecture. All polling is done in the same table and each event of relevance has a trigger to write in this table.

The following subsection will describe how the architecture modules work. There are 4 types, as described previously, and all of them use the database and the file system. We start explaining the main module (Master & Commander) while the other three are instantiated for each model orchestrated.

### 3.2.3. Master & Commander

The Master & Commander is the main module, which means it does exist one and only one instance of this module per execution. It is responsible for initializing the software, getting the initial inputs and starting all simulations. Its basic workflow starts establishing the connection with the database (RDBMS) and reading the topology of the orchestration to be executed together with all the simulation parameters. Then it checks if all models required exist in the file system, if so, the execution moves on to generate the database schema to the corresponding topology, creating the triggers for each module and initialising the file system creating a folder for each set of initial inputs copying all the necessary files to call a simulation.

After this initialization, the Master & commander invokes all the modules necessary for execution asynchronously. Each module will be started in a separate thread and all of them will be active simultaneously during the whole simulation. Having the initialization steps concluded with success, the Master module starts the execution, registering a new orchestration in the RDBMS, a

new simulation referred to the first component model execution and insert a new input for the first model to trigger its serialisation. The Master & Commander polls the DB checking if all initial inputs have a corresponding output. Once this condition is satisfied, all threads are closed and the simulation finishes (Fig. 3.8).

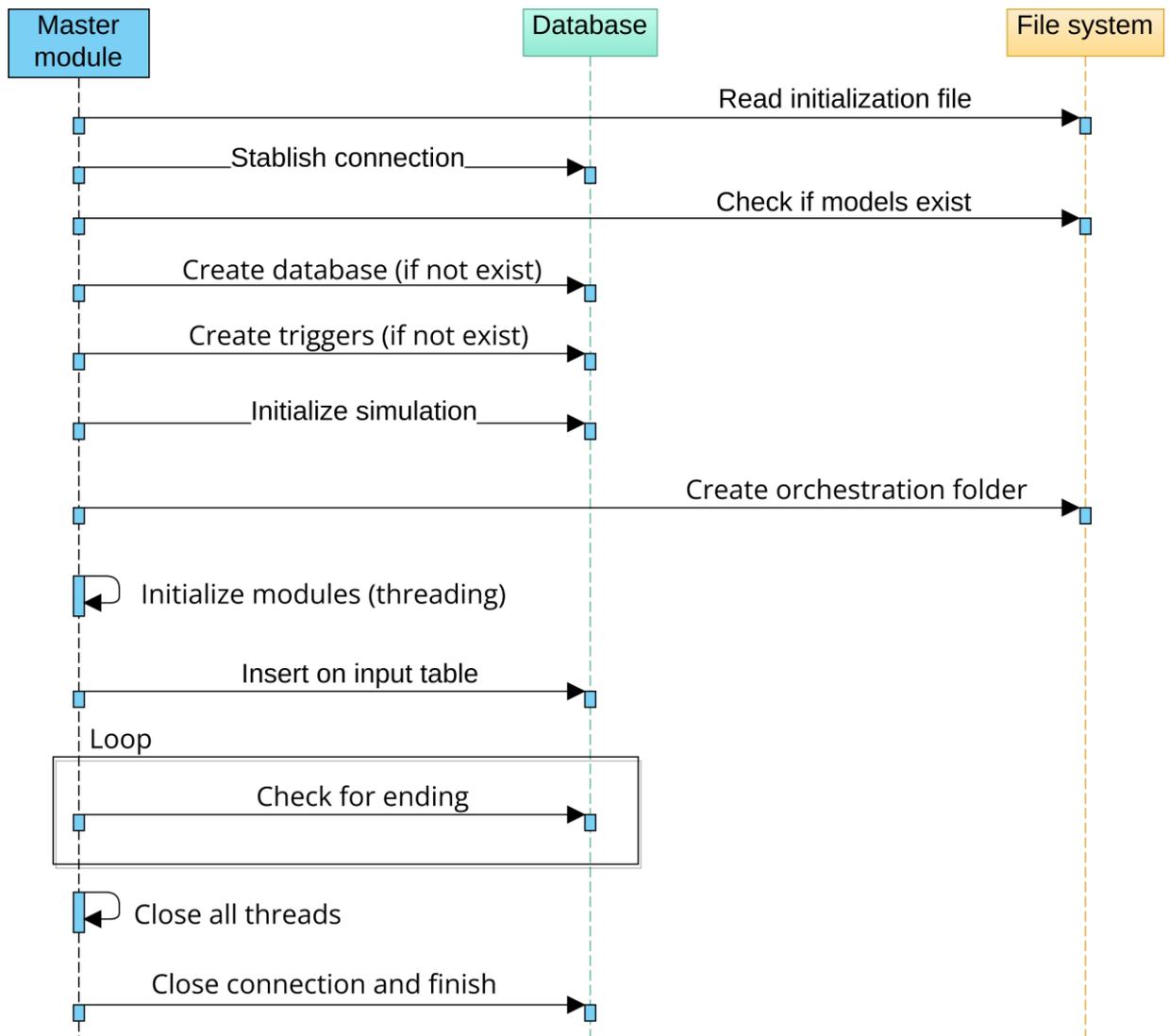


FIGURE 3.8: Master & Commander sequence diagram

### 3.2.4. Serialisation

The Serialisation module oversees the homonym process. A serialisation consists of getting a set of data and writing them in files. Each instance of the serialisation module corresponds to one model and is composed of a serialiser function that is developed specifically for each model type. A model type is defined by the I/O and the call command for the component model.

The serializer module, like any module, starts establishing a connection with the database and gets the model info in order to know which columns of the database it must read and the name of the input files that it will be generated. After this, it polls during the whole execution for new insertions in the respective input table in the database.

Once the polling is satisfied, it runs the serialisation and updates the files that originated in the database. Aiming to improve the orchestrator performance, this module, after the serialisation verifies if this set of inputs is repeated. In an affirmative case, the associated output files are copied

into the same file before restarting the polling (Fig. 3.9). Only the files are copied, in order to not break the rule of each module writing only in one database table, thus keeping the architecture with fewer interdependencies. This output verification is active by default; however, is not recommended if the model time consumption is short or for stochastic models.

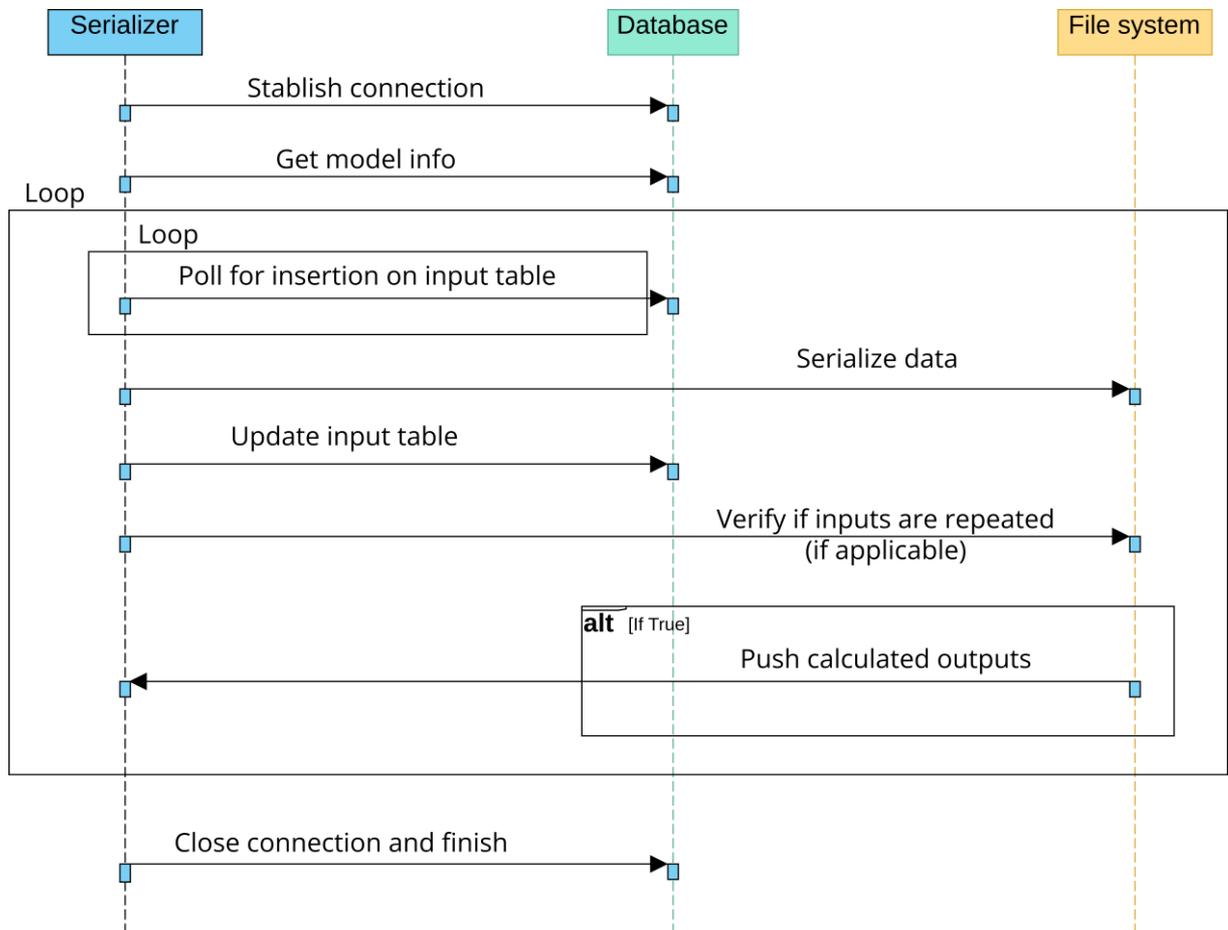


FIGURE 3.9: Serialisation module sequence diagram

### 3.2.5. De-serialisation

The de-serialisation module is responsible for the counterpart of the serialisation process, that is, it takes a set of files and registers them in the database. These files are the outputs of the corresponding model. The de-serialisation module is also responsible for running the model and monitoring its executions. This task was included in this module because the end of a model is registered to do an insertion in the output table in the DB and only this module can insert in this sort of table.

After connecting to the database and getting information about the corresponding model and its outputs, the path to the model is taken in order to copy it to the subfolder where the model will be called. Each model is run in the same folder as the inputs are saved and the outputs are generated in the same directory.

The de-serialisation module polls for an update in the respective input table with the input filenames created. So, it starts immediately after the serialisation module writes them in the database. Then, it is checked if the output files already exist in the path indicated. If so, it means the inputs are already calculated and it is only necessary to register the output in the database.

If the simulation is new, the module checks if the minimum number of inputs is ready to call the model. For most cases, one input is enough for calling the model. However there are situations where multiple executions of the same model can be necessary inside the topology. For these cases, aiming to speed up the tool, it was added the possibility of distributing these executions in multiple jobs. Aiming to facilitate the tracking of these calls and to prevent the orchestrator from calling a large number of jobs, and eventually extrapolating the cluster restrictions, the number of jobs must be known previously. This number is set at the beginning of the simulation, in the initialisation file, and the number of simulations called per job is calculated internally for each interaction. Once the job is called, a few seconds later it is checked if the call was done with success. Although rare, errors out of the orchestrator scope can happen in this step such as API out of service or connection failures or any other problem at the HPC level. In this case, the model is re-called periodically until it works.

After calling models, it is necessary to wait for them to finish. The de-serialisation module checks periodically for the status of the jobs called. All the jobs are registered in the database to preserve its data and to avoid re-calling simulations going for interrupted orchestrator executions.

As jobs can be completed with success, finish with error or be killed, exceptions treatments were necessary. For the first case, a double-check is done to verify if all output files were produced. When an error status is returned, it is verified if the error was caused by `TIMEOUT` or `OUT-OF-MEMORY`. In both cases, the job is re-called allocating double of the time or the memory previously required, until the job finishes for a different reason. When the status refers to a model error or an interruption for any reason, the error is reported as registering an empty output, which will trigger the end of orchestration in the following relation module.

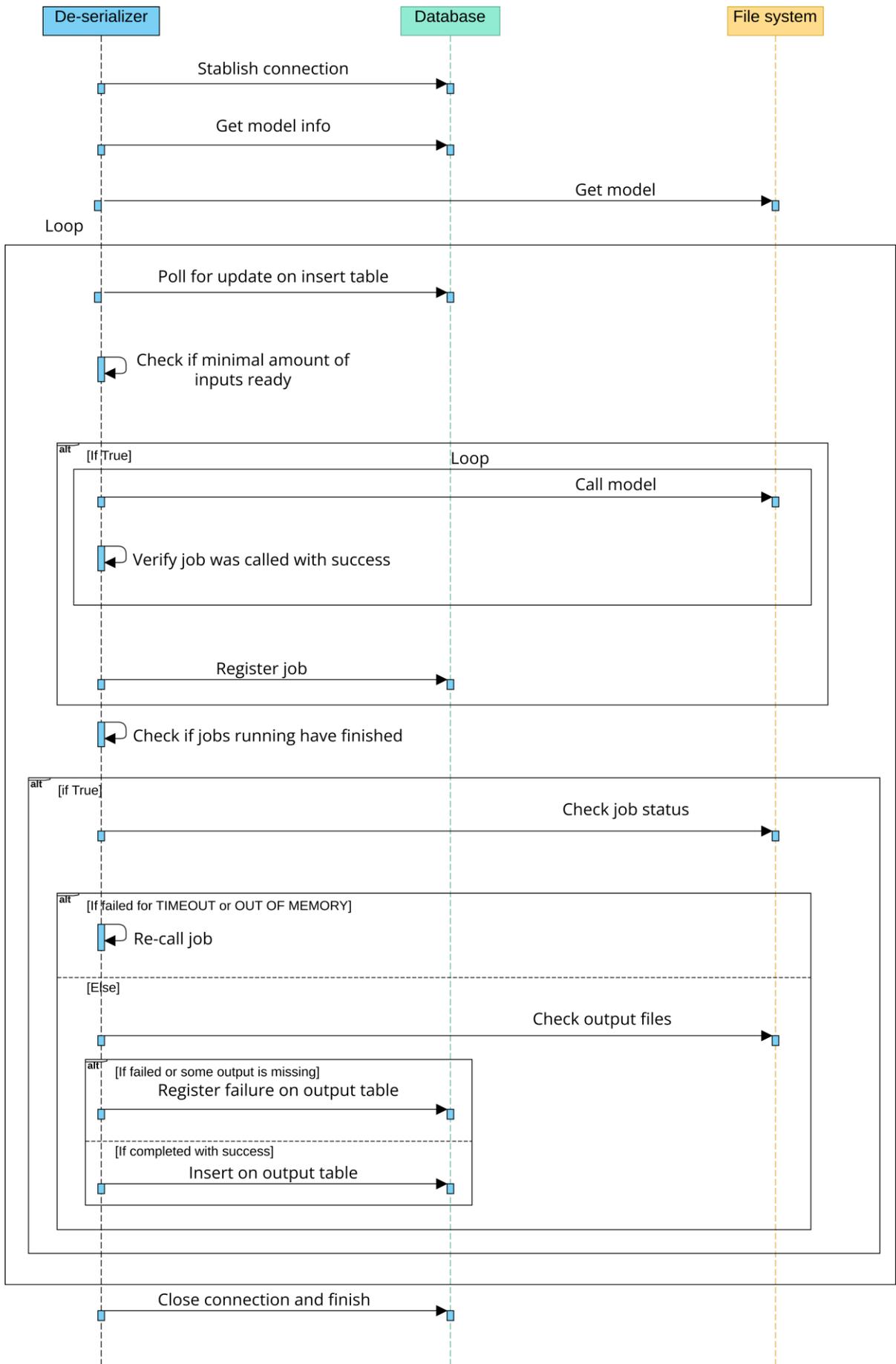


FIGURE 3.10: De-serialization module sequence diagram

### 3.2.6. Relation modules

To transform the outputs of a model into the inputs of another model some data processing is usually required. Such processing can be limited to a transformation of the serialisation format, or it might involve more complex operations such as resampling. This task is done in a *relation module*. The relation modules are similar to the serialisation and de-serialisation modules with the difference that they read from an output table and write in an input table and, also, they don't call jobs, all the processing is implemented locally or by isolated scripts that are executed in a serial mode in the thread. They are not generic; they must be very specific for a pair of model types since the processing between them depend strongly on their I/O files and formats. Therefore, a change in one model implies updates on its relation modules (before and after the model).

A relation module instance establishes a connection to the database and polls for insertion in the output table. Once the polling finds an insertion, the module gets all data necessary for the following model inputs that can come from multiple sources, in the database and in the file system. Afterwards, the file system is prepared for the next model and the data processing run and registers the input data in the DB triggering the beginning of the next serialisation module.

A special case happens if the relationship is the last one before starting a new loop of execution. In this case, the relation module updates the simulation time of the multiscale model and controls if the orchestration has finished. In a positive case, it copies the results to the output table of the orchestration which will trigger the Master & Command to finish the orchestrator execution.

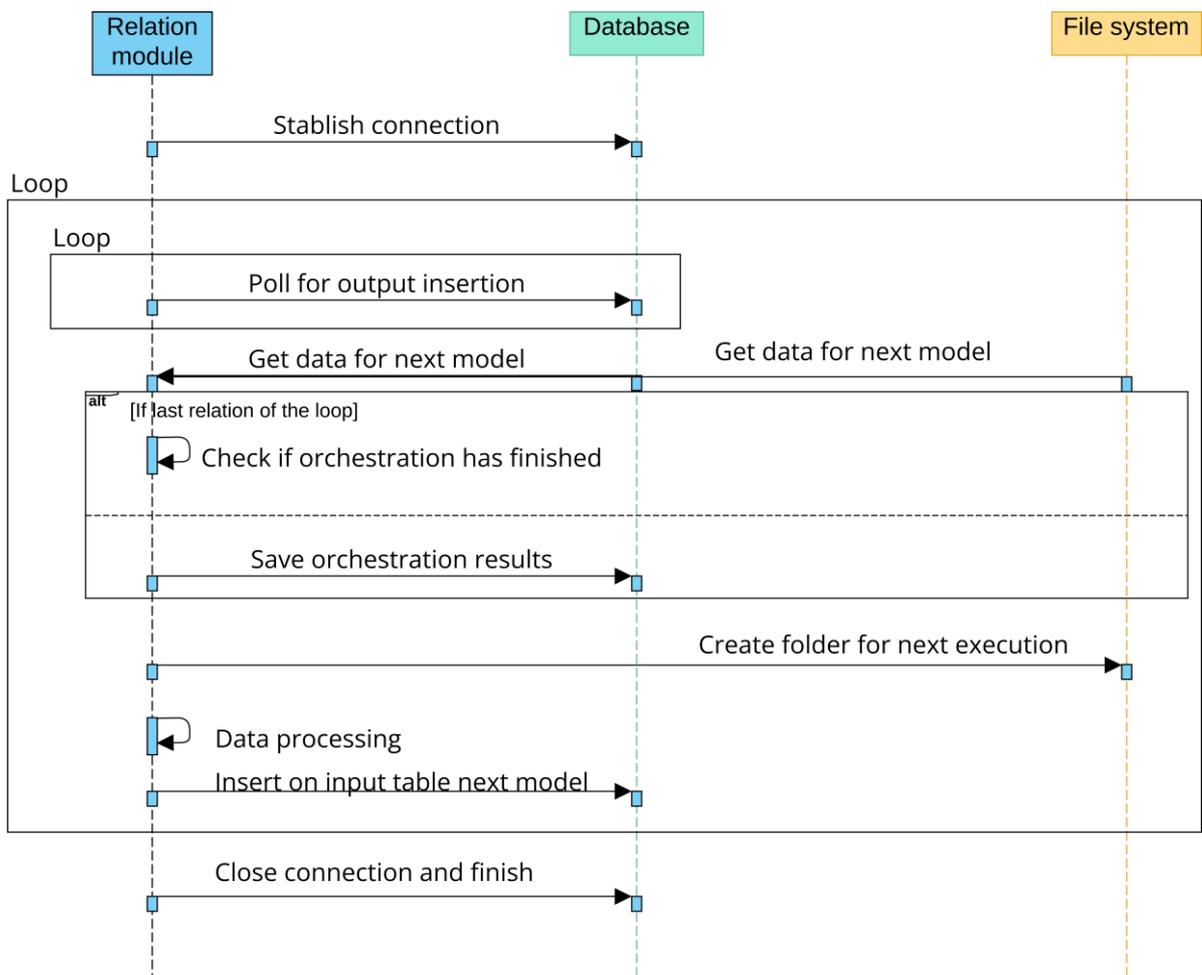


FIGURE 3.11: Relation module sequence diagram

### 3.2.7. Implementation details

The environment of choice for the PRIMAGE implementation was composed of Python3 programming language using MySQL database service in a CentOS7 Linux system due to simplicity, the previous familiarity of the team, and to facilitate the execution of the HPC systems used over the project.

The code was developed using an Object-Oriented structure, as presented in fig. 3.12. There is a class to encapsulate all SQL functions once all modules use them and a Type model class that is instantiated inside the serialisation and de-serialisation modules and isolates all methods developed specifically for a model type (e.g., serialiser, de-serialiser and call model). In this way, an update in a model only requests the creation of a new type of model class and new relation modules with no need to modify the other modules.

The other modules are developed to be the most generic possible; however, at the current stage, minor adjustments are necessary for the serialisation and de-serialisation modules depending on the topology (e.g., activate or not the verification of the repeated input, set if multiple models will run in the same execution) and the initial inputs are hardcoded inside the Master & command.

The framework will be released as an open-source tool in its final version and, likely, will be tested for different multiscale models and adjusted and speeded up.

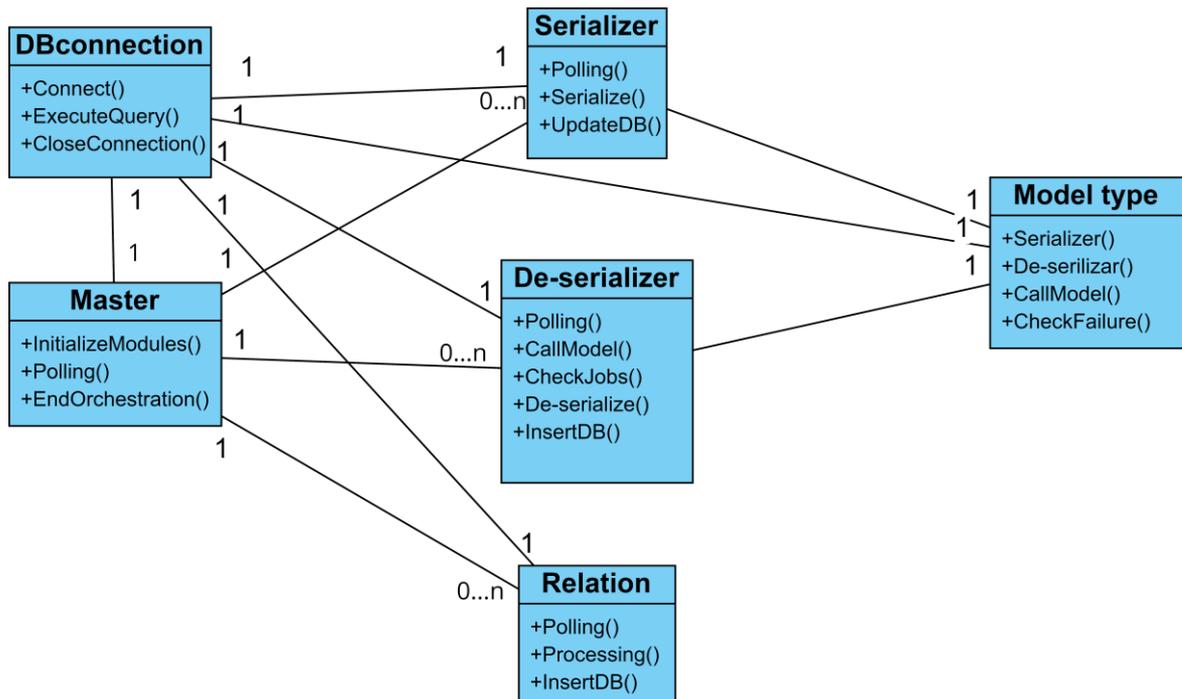


FIGURE 3.12: Class diagram of VPH-HFv3

## 4. Development and validation of a particularization strategy

This chapter reproduces in full the accepted version of the paper “Effect of particularisation size on the accuracy and efficiency of a multiscale tumours’ growth model”, by Varella V, De Melo Quintela B, Kasztelnik M, Viceconti M. International Journal for Numerical Methods in Biomedical Engineering, <https://doi.org/10.1002/cnm.3657>. Following the rules of the publisher, we will publish the paper in Open Access of ARXIV 12 months after its publication. Consistently, the thesis will be published online with a 12 month embargo.

### EFFECT OF PARTICULARISATION SIZE ON THE ACCURACY AND EFFICIENCY OF A MULTISCALE TUMOURS’ GROWTH MODEL

Vinicius Varella<sup>2,3</sup>, Bárbara de Melo Quintela<sup>1</sup>, Marek Kasztelnik<sup>4</sup> and Marco Viceconti<sup>2,3</sup>

1 Departamento de Ciência da Computação - Universidade Federal de Juiz de Fora (BR)

2 Department of Industrial Engineering, Alma Mater Studiorum - University of Bologna (IT)

3 Medical Technology Lab, IRCCS Istituto Ortopedico Rizzoli, Bologna (IT)

4 Academic Computer Center Cyfronet AGH, Krakow (PL)

#### ABSTRACT

*In silico* medicine models are frequently used to represent a phenomenon across multiples space-time scales. Most of these multiscale models require impracticable execution times to be solved, even using High Performance Computing systems, because typically each *Representative Volume Element* in the upper scale model is coupled to an instance of the lower scale model; this causes a combinatory explosion of the computational cost, which increases exponentially as the number of scales to be modelled increases. To attenuate this problem, it is a common practice to interpose between the two models a *particularisation* operator, which maps the upper-scale model results into a smaller number of lower-scale models, and an operator, which maps the fewer results of the lower-scale models on the whole space-time *homogenisation* domain of upper-scale model. The aim of this study is to explore what is the simplest particularisation / homogenisation scheme that can couple a model aimed to predict the growth of a whole solid tumour (neuroblastoma) to a tissue-scale model of the cell-tissue biology with an acceptable approximation error and a viable computational cost. Using an idealised initial dataset with spatial gradients representative of those of real neuroblastomas, but small enough to be solved without any particularisation, we determined the approximation error and the computational cost of a very simple particularisation strategy based on binning. We found that even such simple algorithm can significantly reduce the computational cost with negligible approximation errors.

#### KEYWORDS

Multiscale, oncology, particularisation, homogenisation, modelling

#### 4.1. INTRODUCTION

In most biological problems, biological entities operating at a spatial scale of some microns (e.g., cells) produce clinically relevant effects that manifest at a much larger spatial scale (e.g. tissue, organ,

organism). Same applies for the temporal scale: chemical reactions taking place in a few milliseconds may produce effects observable over years. For example, with reference to the modelling of solid tumours growth, the space-time scales to span are 10-8m to 10-1m, and 107s, and 10-2s (Quintela, et al., 2022). Hereinafter we define *range* of a model, the largest portion of space or time the model accounts for, and *grain*, the smallest portion of it (Bhattacharya et al., 2021). The grain is frequently referred to also as The Representative Volume Element (RVE). A neuroblastoma can reach volumes as large as 4,188,790 mm<sup>3</sup> (a sphere of 20 cm of diameter). The Representative Volume Element (RVE) of the tissue-scale model is equal to the average size of tumour cell ( $\approx 10$  microns). Current models can account for the interactions of around one million cells (which correspond to a range of  $\approx 1$  mm<sup>3</sup>) over two weeks in roughly 2 minutes of computations, using a single Graphic Processing Unit (GPU). Even in the unrealistic assumption that the parallelisation of such model scales linearly, to solve the tissue-scale model over the entire tumour volume would thus require 111,7011 GPU-hour on a large GPGPU cluster to simulate a full four-months chemotherapy cycle. Splitting the problem into two single-scale models does not significantly change the computational cost: the whole tumour model would need an RVE of 1-2 mm<sup>3</sup>, so we would need to solve millions of tissue-scale models, one for each RVE of the tumour-scale model. However, if we can interpose between the two single-scale models particularisation / homogenisation operators, the number of tissue-scale models to run can be reduced. This comes at the price of introducing some additional approximation error in the solution. It is also important that this operator is as simple as possible, to reduce the computational overhead that the multiscale orchestration imposes.

The simplest particularisation strategy is based on binning. In meteorology, the Pairwise Homogenization Algorithm uses statistical analysis to dynamically define the bins (O'Neill et al., 2022). In fracture mechanics, binning-based particularisation is implemented by placing bins at the Gauss quadrature points (Huq et al., 2016). Binning particularisation is also used successfully in the reconstruction of genomes (Celis et al., 2018), but to authors' knowledge, it has never been proposed for computational oncology problems. There are of course more sophisticated particularisation methods described in the literature. In fracture mechanics are common hybrid multiscale methods combining homogenization and domain decomposition approaches e.g., (Vernerey and Kabiri, 2014). In the field of composite materials, it is common the use of the theory of asymptotic particularisation of periodic media (Babuška et al., 2014; Engquist and Souganidis, 2008), which can be formulated with guaranteed accuracy (Paladim et al., 2017). Similar methods were also used to homogenise the properties of solid tumours (Collis et al., 2017); however, two key assumptions (periodic microstructure, and strong separation of scales) are hardly met in this type of tissues. Problems of mass transport by diffusion within composite materials have been homogenised assuming that the mass release curves for the detailed microstructural and continuum models (Kojic et al., 2014). De la Cruz *et al.* propose a quite elegant hybrid method to homogenise continuum and molecular-scales in tumour growth multiscale model (de la Cruz et al., 2017). However, their purpose is not that of reduce the computational cost of the multiscale model. On the contrary, the method they propose introduces a non-negligible overhead to homogenise the boundary conditions between the two scales.

The aim of this study is to evaluate the error caused by a specific implementation of the particularisation / homogenisation process based on binning, as a function of the number of bins, in a multiscale model of growth for neuroblastoma tumours. The model uses the particularisation / homogenisation process to link a whole-tumour scale continuum model that calculates oxygen diffusion and the biomechanical interaction of the growth process, with tissue-scale agent-based models that simulate the activation, replication, differentiation, and death of the various cellular populations involved with this specific tumour growth. As the overall computational cost of the model's solution depends on the number of tissue-scale models we need to run at each time step, which is equal to the number of bins used for the organ-to-tissue particularisation, there is a trade-off between the level of detail of the particularisation and the computational cost to solve the multiscale model.

## 4.2. MATERIALS AND METHODS

### 4.2.1. A Brief overview of the biological problem and its idealisation

One of the aims of the PRIMAGE project (Martí-Bonmatí et al., 2020) is the development of digital twins (patient-specific models) that can predict the growth of a neuroblastoma (a type of solid tumour) when left untreated or when treated with different chemotherapies. Living cells can replicate through a process called mitosis, where a single cell divides in two. Healthy human cells have a number of molecular mechanisms that limit the rate of replication, and the number of times a cell can replicate during its life. Due to mutation some cells might lose these limitation mechanisms and start replicating indefinitely. This produces a solid tumour, which is a tissue mass composed of tumour cells and of the extracellular matrix they secrete. As the tumour grows it compresses the surrounding tissues and organs, compromising their functions and eventually killing the patient. There are various things that may slow down the growth of a neuroblastoma, but the PRIMAGE model focuses on three: the transport to and from the surrounding vascular network that brings oxygen and other nutrients to the growing number of cells and removes from them their metabolic waste substances; the biomechanics of volumetric expansion of the tumour geometry due to the cells' replication; how the cells respond to the specific chemotherapy. Oxygen diffusion in the tumour mass is regulated by the *cellularity* (the volumetric ratio between cells and matrix) and the *vascularity*, the density of capillary vessels that form within the tumour as it grows. Chemotherapy may reduce the replication rate and increase the death rate of tumour cells. How effective a specific chemotherapy cocktail is in doing this depends on molecular make-up of the mutated tumour cells.

The PRIMAGE model predicts the change in geometry of the tumour over time depending on the treatment, as a function the following input set: geometry, cellularity, and vascularity at the beginning of the treatment, which are quantified with Magnetic Resonance Imaging; and several molecular biomarkers obtained by analysis the tumour biopsy. The mathematical model is built assuming that each cell change states is a probability function  $\pi_{\gamma_k}(I_k, \alpha_k, \tau_k, S_1, \dots, S_J, t) \cdot \pi_{\gamma_k}^{treat}(T_l)$  that depends on the type of cell  $I_k$ , its differentiation level  $\alpha_k$ , its telomerase<sup>5</sup> state  $\tau_k$ , and the concentration biochemical species  $S_i$  at the cell location.  $\pi_{\gamma_k}^{treat}$  is the cumulative probability of internal state change for cell  $k$  as a function of the treatment type  $T_l$ . The changes over space and time of the concentrations  $S_i$  can be formulate as a diffusion-reaction equation. The density of the various cell types in space as a function of the replication/death events and of the biomechanical deformation of the tissue can be formulated in terms of mass conservation equations. If we assume the cellularity does not change as much as the tumour grows, the change of volume of the tumour over space and time can then be formulated as a partial differential equation function of the changes in the local concentration of cells. The complete mathematical treatment can be found in (de Melo Quintela et al., 2022).

Scale separation analysis (de Melo Quintela et al., 2022) suggested to decompose the problem into three single-scale models, properly orchestrated. The cell model, which computes  $\pi_{\gamma_k}^{treat}$ , needs to be run only once at the outset. Thus, the orchestration is limited to the coupling of the tumour model with the tissue model. With respect to the results of the scale separation analysis, the final implementation of the multiscale model presents some differences, dictated by computational constraints. The tumour model solves the diffusion-reaction equations for the initial conditions dictated by the imaging data using a finite element scheme. A tissue model is then run for each finite element: the tissue models simulate the evolution of the cells in the compartment for the following 14 days. This is the longest time that in the untreated model predicts a change in volume that the tumour model can handle without the need for remeshing. With the updated values of volume, cellularity, and vascularity for each finite element the tumour model simulates first the volumetric expansion simulation. The resulting geometry is the re-meshed, and then the diffusion-reaction

---

<sup>5</sup> Telomeres are a portion of cell DNA placed at the extremes of the chromosomes. Every time a cell replicates it telomerases shorted; this poses a limit to number of replications a healthy cell can undergo. Telomerase is a protein that extend the telomeres, extending its ability to replicate, as it happen in tumour cells.

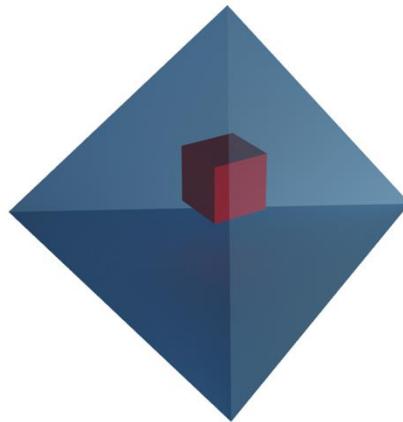
simulation is run again with the new values of vascularity and cellularity. The tissue scale models are executed again to simulate another 14 days, and so forth until the whole chemotherapy cycle (typically 16 weeks) is simulated.

#### 4.2.2. Implementation details

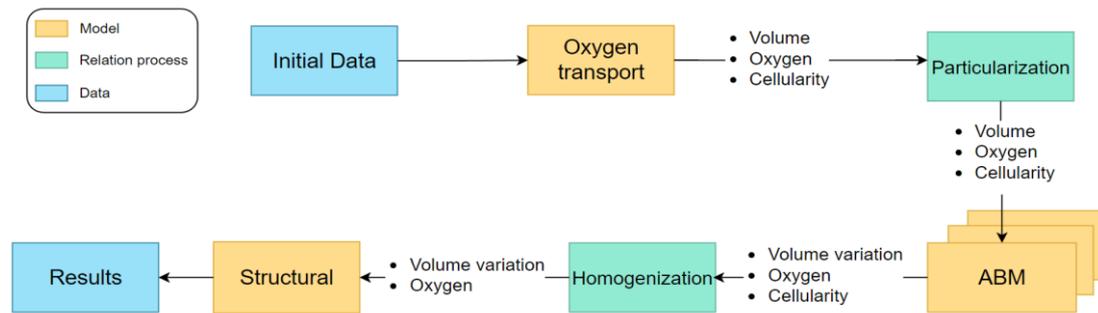
The current version of the PRIMAGE multiscale model requires the coupling of two component models. The first component model is a finite element model developed by the University of Zaragoza (ES) that solves the diffusion-reaction equations and the biomechanics of the volumetric expansion equations. The volume of the whole tumour at time zero is decomposed into a finite element mesh of 4-node linear tetrahedrons. The average element size is an order of  $10^{-2}$  mm. The tumour-scale model is entirely deterministic; if the model is run twice with the same input, it will predict the same output. The tumour-scale model is solved using a commercial general-purpose solver (Ansys v19.5, Ansys Corp, USA), which runs on CPU cores.

The second component model is an agent-based model (ABM) developed by the University of Sheffield (UK). In the ABM tissue-scale model each relevant cellular type is modelled as an autonomous agent, whereas the diffusion-reaction of the chemical species in the compartment are described through a system of differential equations solved with a finite difference scheme. The growth of the tumour is described by a set of rules that regulate when each cell moves, replicates or dies. The tissue-scale model is stochastic in nature, so the same input will not produce the same output. The ABM model is run using the Flame GPU framework (Richmond et al., 2010), also developed by the University of Sheffield, and distributed in Open Source (Richmond et al., 2021). The ABM runs predominantly on GPU cores.

The Reference Volume Element (RVE) for the ABM is a cube volume shrunken based on the initial parameters of the model (e.g., element volume, cellularity, and cell density). We assume that the ABM RVE is fully contained inside the tetrahedron of the tumour-scale model and positioned at its centroid (fig. 1). The two models are coupled by an orchestration software layer that handles the data flow, according to fig. 2. Between the two models there is a relation process, described in the following section, that handles the scale transformations.



**Figure 1:** Cubic-shaped RVE for the ABM (red) for a tetrahedron mesh element.



**Figure 2:** Topology of the orchestration software layer

In principle, we should run a tissue-scale model for each finite element in the tumour-scale model. But considering that real tumours models could have millions of elements, this is not possible. Thus, a relation model is added to the orchestration to handle the particularisation / homogenisation process.

The multiscale model was run on PLGrid<sup>6</sup> Prometheus HPC cluster<sup>7</sup>, managed by Cyfronet, composed of Intel Xeon E5-2680 CPUs and 144 Nvidia Tesla K40 XL GPGPUs. The cluster uses Linux CentOS7 as an operating system and the SLURM scheduling system.

#### 4.2.3. Data

The whole study was conducted on an idealised tumour model, small enough to allow a solution without any particularisation with the computational resources available. The idealised tumour model assumes the cancer to be a spherical solid tumour of 5mm in diameter, an initial volume of 524 mm<sup>3</sup>, which is meshed with 4448 finite elements. Vascularity is represented in the model in term of oxygen concentration. Oxygen concentration is dimensionless because is normalized with maximum oxygenation levels according to literature; the initial oxygen concentration is assumed to be 1 at the outer skins and 0 at the centre with linear variation along the radius. The produce an initial gradient that equal or greater the one we would expect in real tumour data. The model assumed the cellularity (ratio between cancer cells volume and the total cancer volume) to remain constant over the entire simulation. The multiscale model is run to reproduce the growth of this idealised tumour for a period of two weeks, without any treatment. The treatment would always slow down the growth, so the no-treatment option is the most critical for the particularisation algorithm. While, ideally, we should calculate the particularisation error over the full duration of the chemotherapy cycle, as the error propagates, the re-meshing required after 14 days would make impossible a reliable evaluation of the particularisation error.

#### 4.2.4. The particularisation/homogenisation relation model

Between the two component models there is a relation model that has the function of gathering all the data necessary to write the inputs of the next model and doing the transformations requested for making the scale change, here named as particularisation and homogenisation (fig. 2).

Those processes complement one with each other, the particularisation gets the outputs from the macro-scale and groups the elements of the mesh in sets called bins. The homogenization gets the outputs of the micro-scale and estimates them to the macro-scale.

<sup>6</sup> <https://www.plgrid.pl/en>

<sup>7</sup> <https://www.cyfronet.pl/en/computers/15226,artykul,prometheus.html>

The binning process algorithm orders the elements by one variable and groups them in sets of the same size (or the closest possible to it). The biggest value of each bin is selected to run the tissue model.

The homogenisation gathers all the results obtained by the tissue model and estimates the values (oxygen and volume) for the elements that were not run, using a linear approach on the order the elements were sorted at the particularisation, resulting in a set of data by elements. However, the last component model requires that the oxygen input values are written per each node of the FE and not per element. This is done by considering the concentration of oxygen in each node as a simple average of the oxygen value of all the elements that are connected to the node.

The particularisation algorithm can operate on one variable at a time. In our case, the coupled models exchange two quantities, the oxygen concentration, and the change in volume. A preliminary investigation confirmed that the spatiotemporal gradients of oxygen concentration were much greater than those of the changes in volume; Thus, the particularisation algorithm was run on the oxygen concentration. However, we also monitored the error induced in the other variable. On the contrary we did not explore the effect of particularisation on cellularity, because in this implementation it is assumed to remain constant.

#### 4.2.5. *The validation study*

We first run the whole multiscale model without any particularisation (number of bins = number of finite elements). At the end of the simulation, we recorded for each finite element of the tumour-scale model the oxygen concentration and the change in volume (tumour growth).

We then repeated the simulation several times, each time progressively reducing the number of bins. Since the tissue-scale model is inherently stochastic, if we rerun the model, we would see differences not only due to the number of bins we use but also due to the stochasticity. In order to separate these two sources of variation, in each of these reruns, we did not run again the tissue-scale model, but we simply used the full-resolutions results obtained with the initial simulation. This way, we were certain that the only difference between repeated simulations would be due to the level of particularisation.

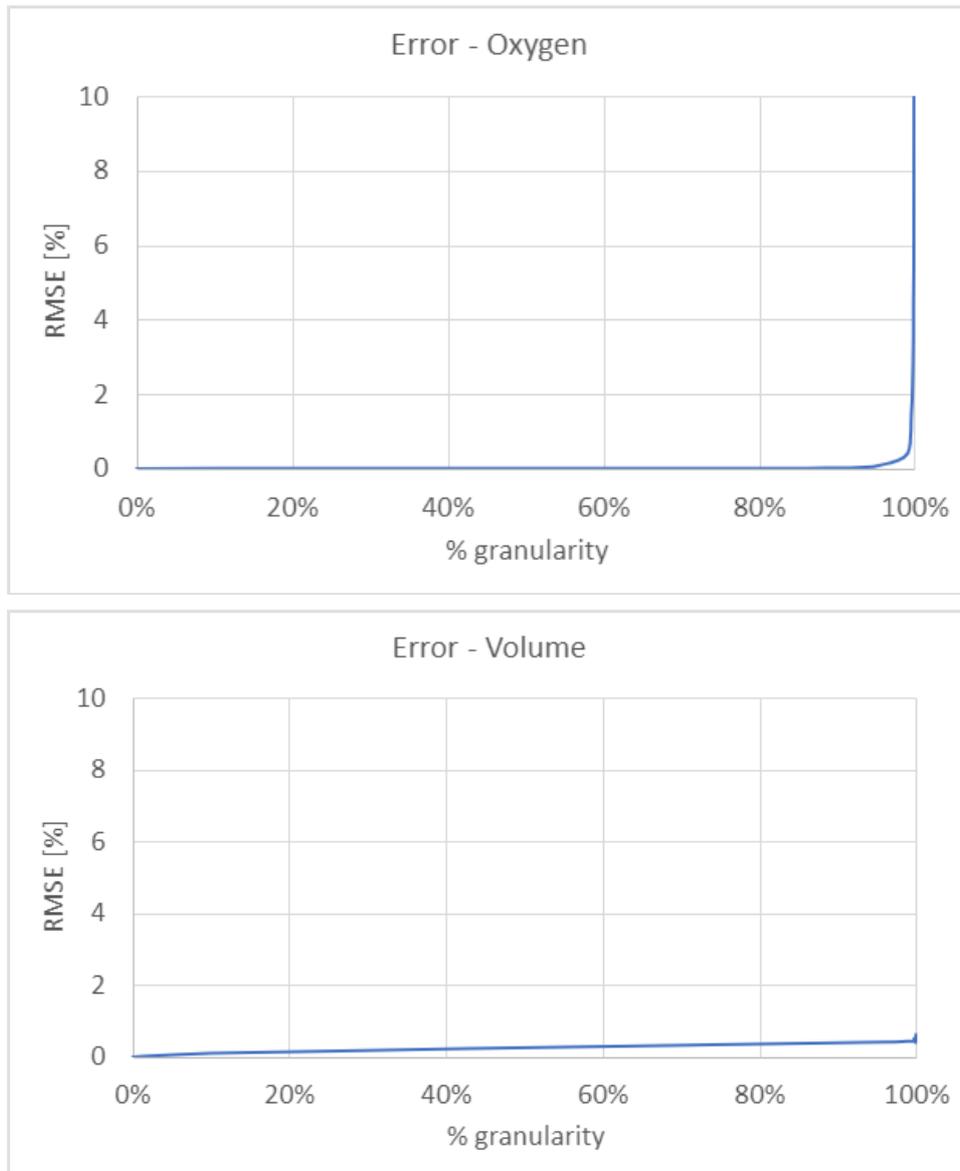
To describe the average error caused by each level of particularisation for each coupled variable (oxygen concentration, and changes in volume), we used the Root Mean Square Error (RMSE). The RMSE was plotted as a function of the % of tumour-scale elements that were simulated at the tissue-scale level (hereinafter referred as *granularity*, defined as the opposite of particularisation).

Considering the primary reason for particularisation is to reduce the duration and cost of the simulation, we also plotted the RMSE as a function of the wall-clock solution time, and of the total number of core-hours (under the assumption that CPU and GPU cores had the same weight).

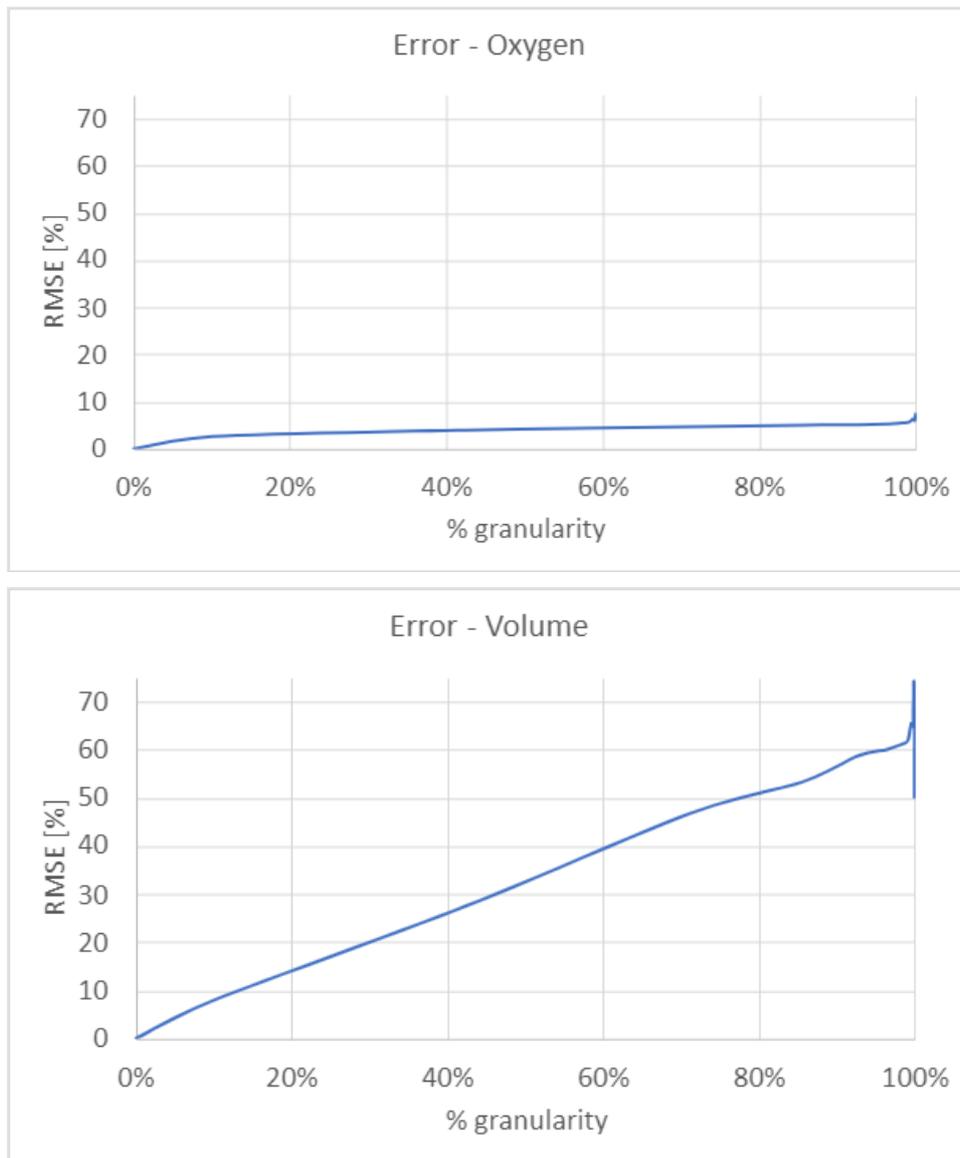
### 4.3. RESULTS

Two cases of particularisation were tested, particularising by oxygen concentration and by the variation of volume. In each test, several executions were run only changing the size of the particularisation (number of bins) and comparing to the value obtained without doing particularisation and homogenisation (full resolution result) for calculating the Root Mean Square Error (RMSE), divided by the full resolution result to normalise. The % granularity represents the fraction of finite elements that had the outputs estimated by the tissue-scale model, that is, the percentage of granularity is equal to one minus the percentage of elements particularized. Therefore, the smaller is the % granularity, the closer it is to the case without particularisation (which has 0% granularity).

The %RMSE error is plotted vs the % granularity for both variables where the particularisation variable is the oxygen concentration (fig. 3), or the variation of volume (fig. 4).



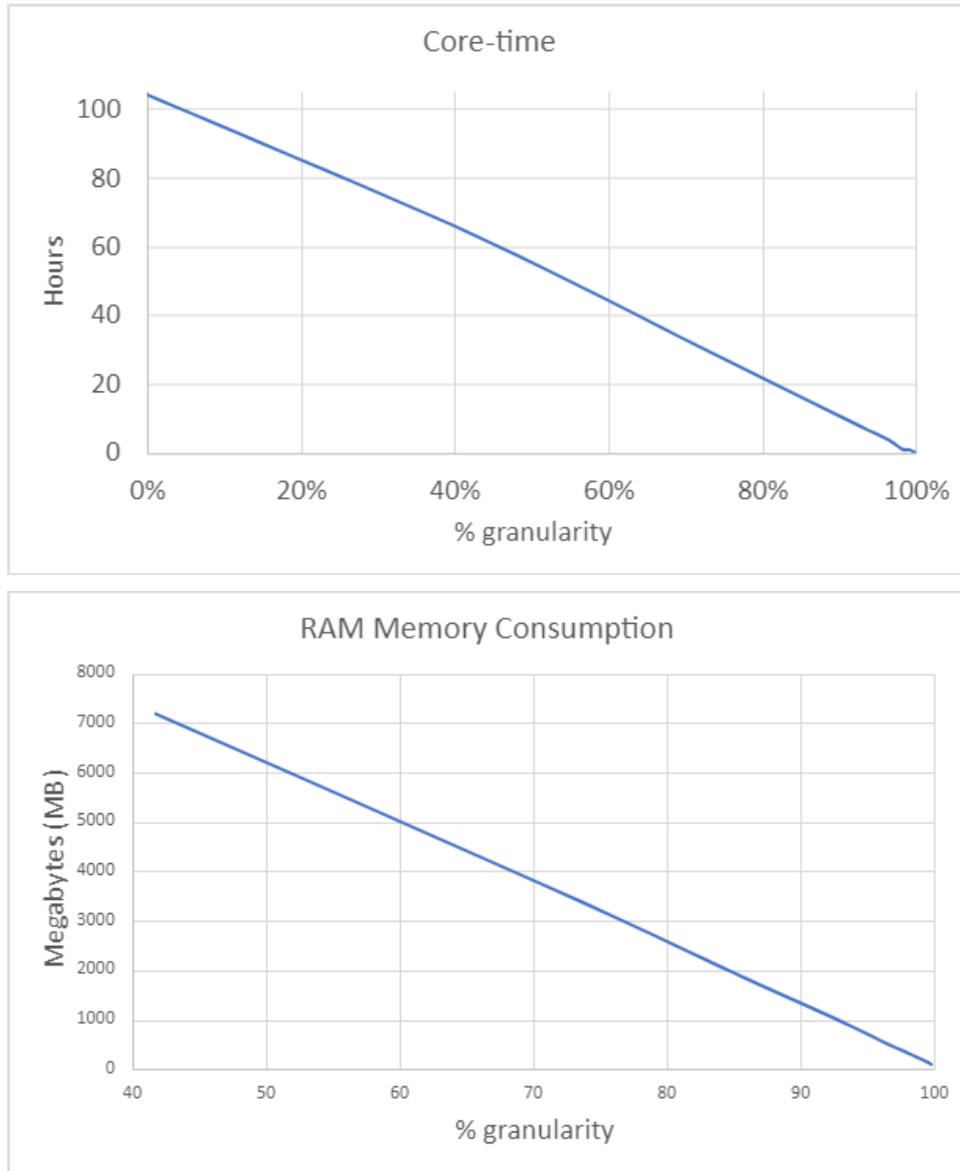
**Figure 3:** Error of the results for the oxygen concentration and volume variation with particularisation by oxygen.



**Figure 4:** Error of the results for the oxygen concentration and volume variation with particularisation by volume.

The particularisation by oxygen resulted to be much more accurate. While the particularisation by volume at 95% granularity cause errors of 60% on the variation of volume and 7% in the oxygen concentration, the particularisation by oxygen at 95% granularity caused errors below 1% for both variables (specifically 0.4% for the variation of volume, the primary output of the model).

The computational cost for the whole orchestration is reported in figure 5 in terms of core-time and memory allocation for the case of particularisation by oxygen. While the simulation of two weeks of growth for a small, idealised tumour used in this study requires 104 core-hours and over 14 GB of allocated memory to be solved on the in the Prometheus cluster without particularisation, an 95% granularity model can be solved with only 5 core-hours and 700 MB of memory. This means that the largest tumour would require 400,000 core-hours to simulate the whole four months of chemotherapy for each of the treatments being tested. Considering that current pre-exascale European supercomputers like Leonardo being installed at the CINECA Italian HPC centre has  $\approx$  14,000 GPU cores that are nearly ten times faster than the one we used in this study, a simulation could be done in 8-10 hour with a third of the core available.



**Figure 5:** Core-time consumption varying the granularity of the particularisation.

#### 4.4. DISCUSSION

The aim of this study was to evaluate the error caused by a specific implementation of the particularisation / homogenisation process based on binning, as a function of the number of bins, in a multiscale model of growth for neuroblastoma tumours.

As expected, the binning-based particularisation operator introduced approximation errors that grew with the granularity (number of bins) used in the particularisation, while the computational cost showed an inverse linear correlation with the granularity. Thus, the question is reduced to whether the errors caused by significant granularity (e.g., > 90%) introduction approximation errors that are considered acceptable.

In both cases of particularisation, as expected, the error tends to zero when the granularity tends to 0% because fewer values are being estimated (figs 3 and 4). A particularisation by oxygen concentration with less than 1% of elements sampled (99% of granularity) causes a particularisation

error of less than 1% for both oxygen concentration and change in volume. The particularisation by the change of volume is less effective, causing a particularisation error for the oxygen higher than 1% in most of executions and the volume error significantly increases when the granularity is not close to zero.

The core-time consumption graph shows that the impact of the particularisation on the performance is linear and the same occurs to the memory consumption. This occurs because the number of bins determines the number of executions of the tissue model, which is the most computer-demanding part of the orchestration.

This brings us to conclude that, for this problem, a particularisation by binning, while the simplest possible strategy, appears to be adequate. In particular, the particularisation by binning of the oxygen concentration cause approximation errors of less than 1% over a two-week simulation. Considering that the minimal clinically important difference in evaluating a solid tumour treatment is a reduction of at least 35% of the tumour volume (Beaumont et al. 2015), even assuming a full accumulation of the error the error for a four-month simulation would still be well below 5%, which can be considered acceptable for the application at hand.

The computer modelling of neuroblastoma growth has received attention in the literature, (e.g., (He et al., 2018; Kasemeier-Kulesa et al., 2018)), but most models investigate the problem at a single scale. On the contrary, there very little literature to compare to for multiscale modelling of neuroblastomas. In authors' knowledge the only other work (Merla et al., 2019) has very different aims, and use no particularisation strategy because they explore only five possible cellular configurations. If we broaden our research to solid tumours in general, probably the closest work is that done by the CHIC project led by Prof Stamatakos on the modelling of a brain tumour, glioblastoma (Stamatakos et al., 2014; Stamatakos and Giatili, 2017). The project used as orchestration software the precursor of that used in this study (Viceconti et al., 2018), but used no particularisation strategy. Other authors proposed for similar problems single-scale models coupling the cellular replication simulated with a cellular automata with the diffusion-reaction problem simulated with a Lattice-Boltzmann scheme (Alemani et al., 2012). While elegant, this approach, in the paper used to model 1 mm<sup>3</sup> of tumour, is impractical to model tumours that can grow as large as some ten centimetres of size.

The main limit of this study is the use of an idealised tumour model. A real-world tumour would be different from this idealised one, for the sheer size, that might be much larger in some case. However, in term of oxygen gradients, the one we assumed in the idealised tumour model are close to the highest observed in real tumours. A larger size would increase the absolute values of the computational costs but would not change the conclusions on the particularisation errors, which largely depend on such gradient. So, again, the conclusions reached here should remain valid for real-world tumours. Nevertheless, we will repeat a particularisation convergence test for each of the cases used in the future validation studies, in order to confirm that the particularisation error change asymptotically as the granularity is decreased. Another limit is that in the current implementation the cellularity is assumed to remain constant. This should change in the final implementation, and at that point, it will be worth estimating the particularisation error also for that variable. But again, given its spatiotemporal gradients are expected to be much lower than those of the oxygen concentration, we expect comparable or lower errors.

In conclusion, the use of homogenization based on a binning strategy in a multiscale model of solid tumour growth can reduce the computational cost by 90% or more, while causing a particularisation error of less than 1%.

## CONFLICT OF INTEREST

This study was supported by the European Commission through the H2020 project “PRIMAGE: PRedictive In-silico Multiscale Analytics to support cancer personalized diaGnosis and

prognosis, Empowered by imaging biomarkers” (topic SC1-DTH-07-2018, grant ID 826494). MV was also supported by the European Commission through the H2020 project “CompBioMed2: A Centre of Excellence in Computational Biomedicine” (topic INFRAEDI-02-2018, grant ID 823712).

The authors declare that they do not have any financial or personal relationships with other people or organisations that could have inappropriately influenced this study.

## 5. The PRIMAGE orchestration

### 5.1. Description of the final implementation

#### 5.1.1. The component models and the orchestrator layer

The PRIMAGE orchestration starts with running the Image processor, in development by UNIZAR partner. Then it initiates a cycle of four models: oxygen transport (UNIZAR), subcellular model (CHMT), Agent-Based Model (USFD), and biomechanical model (UNIZAR) (Fig. 5.1) as described in detail the chapter 2 (Fig. 2.8). The orchestration is connected to two databases, GIBI and QUIBIM. The GIBI database provides the image data to be pre-processed before running the UNIZAR models, and the histology data used by ABM for calculating several variables internally. The QUIBIM database provides the mutation data and data related to the drug. The models receive some initial and static data from the DBs as presented in (Fig. 2.3), and then exchange data between them. The description of the I/O of each model can be found in table in the appendix 1.

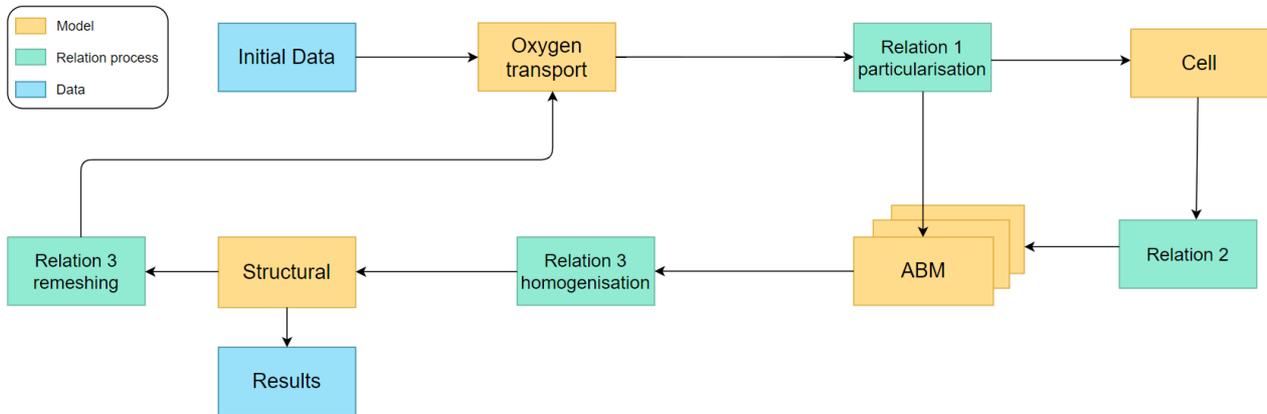


FIGURE 5.1: PRIMAGE topology

The image processor does a pre-processing task, working on images from GIBI image database and returning a mesh of a tumour shape and the initial values to the oxygen transport model and the Agent-Based Model (ABM). The pre-processing, in theory, should be out of the orchestrator since the orchestrator initial inputs represent the inputs of the models. Because the impossibility of running it in the PRIMAGE platform, before the orchestrator, the pre-processor script was inserted into the orchestrator. The orchestrator does not manage the inputs of the pre-processor model. These inputs are communicated to the script by a JSON file prepared by the PRIMAGE platform.

The oxygen transport model computes the diffusion of oxygen through the whole tumour geometry. The model receives the geometry files and the oxygen concentration and vascularization ( $K_{trans}$  value) of each element as input and returns the updated oxygen concentration and the volume for each element of the mesh.

In the whole orchestration, the oxygen concentration is computed as adimensional value scaled by 72mmHg while the volume is in cubic millimetres for the tumour scale models. The following step will select a sample of elements in a process called particularization (described in Section 4.2.4) in order to run the subcellular model and the ABM. The subcellular model estimates, for this sample of elements, the drug inhibition probability on the proteins based on the drug regime previously defined. The model works on the subarray of elements using the vascularization value to return a subarray of outputs that will be consumed by the ABM model. The ABM simulates the multicellular phenomenon, calculating the interactions between neuroblasts and Schwann cells in a patch of the tumour (element). This model is run once for each element sampled. The model receives information of the patch (e.g., volume, oxygen concentration, cellularity etc.) and static data from

GIBI (image DB) and QUIBIM (Genomics DB) databases (Fig. 2.3), like histology, differentiation grade, and some mutations data.

Since the genomics information is not available for every patient, in case it is missing the orchestration will assume the mutation of interest is off. At the first execution loop, the model determines a list of variables (e.g., telomer length, death signals, etc.) by the histology data that need to be received at the following iterations and updated as inputs. After all the ABM executions finish, the homogenization process begins. This process will bring the orchestration back to the whole tumour scale, using the values obtained for the sample of elements to estimate all the values for the other elements. The homogenized data will be used to run scripts that will generate the input files for the structural model and some input files for restarting the cycle (if this is the case).

Given those inputs, the structural model will run for the whole mesh using the volume variation and the properties of the material to calculate the new geometry and volumes. At this point, there is a verification to check if the time of simulation of the orchestration was reached or not. In a positive case, the structural model outputs the results of the multiscale model. Otherwise, a new cycle starts, doing a remeshing and changing the non-static initial inputs from the database for inputs generated by the outputs of the models.

The relation modules cover script executions and data generation to prepare the data for the next model. In figure 5.1, relations 1 and 3 are responsible for the processes of particularisation and homogenisation respectively already described in chapter 4. The relation 2 does a simple process of reading the cell model outputs and separating them into multiple ABM inputs. The relation 3, besides the homogenisation, also runs a script to interpret the homogenised volume ratios from the ABM transforming them into a new format and calculating the elastic properties of the tissues based on the cellularity values (Fig. 5.2).

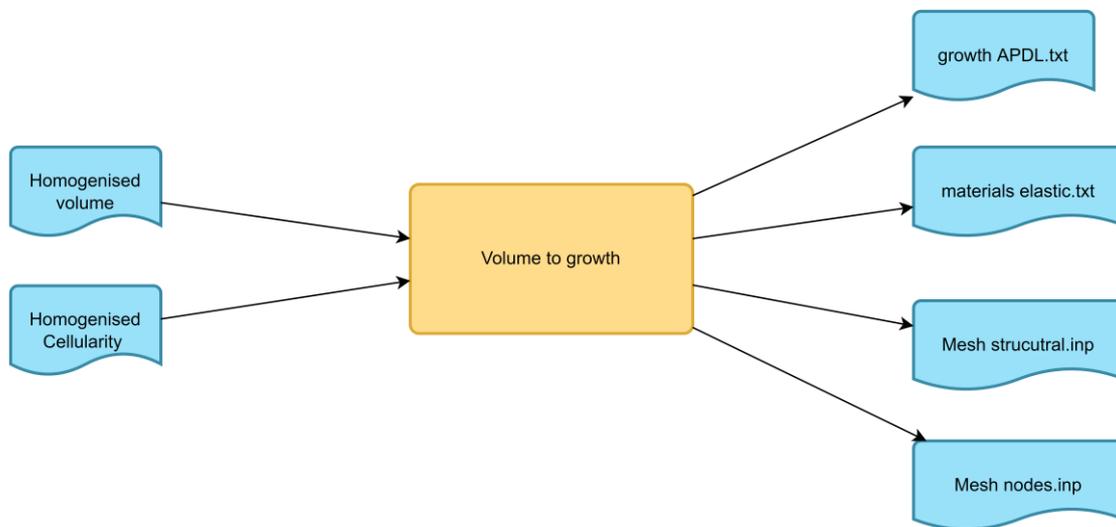


FIGURE 5.2: Processing script inside relation 3 (between the cell model and the Agent-Based model)

Relation 4 hides several processing steps. It starts by getting the structural model outputs and the current mesh to do the remeshing script which will generate an ANSYS model that will produce the new mesh files. Then a third script is run to update in the new mesh all the parameters necessary for the next interaction (Fig 5.3). This relation instance also includes a script that is always run before the oxygen transport model to get all the elemental values and estimate them in nodal values, once the elemental values were more probable to cause errors in the oxygen transport model (Fig. 5.4).

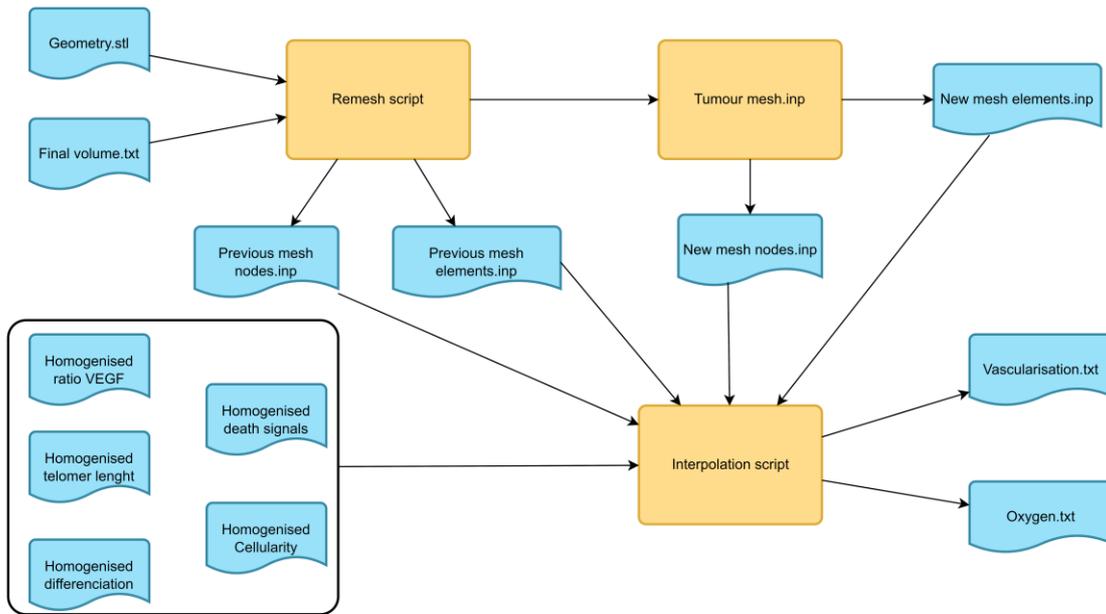


FIGURE 5.3: Remeshing relation module scripts

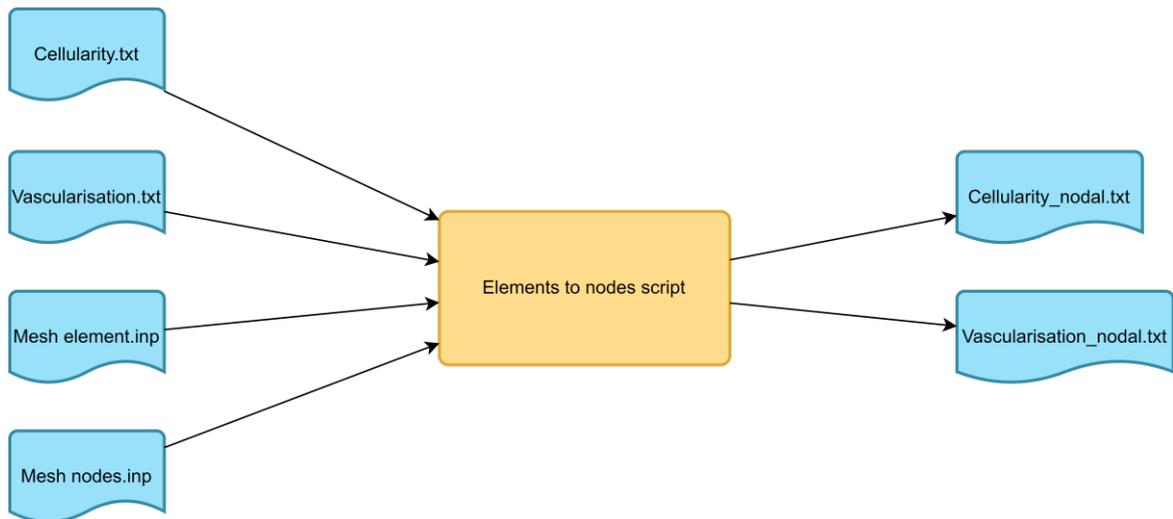


FIGURE 5.4: Script used to run all

### 5.1.2. The HPC platform

We profiled the pilot orchestration throughout the project on the HPC systems provided by the partner Cyfronet. The Prometheus cluster runs Linux CentOS 7 on an HP Apollo 8000, HPE ProLiant DL360 Gen10, with Intel Xeon (Haswell / Skylake) processors, for a total of 53,604 computing cores and 282 TB of RAM. In addition, the cluster has a GPGPU subsystem with 144 Nvidia Tesla V100. The finite element solver used by UNIZAR (Ansys), runs on CPU nodes, whereas the USFD tissue model runs on GPU nodes.

The Ares cluster also runs Linux CentOS 7 on with Intel Xeon Platinum processors, divided into three groups: 256 servers with 384 GB of RAM each, 532 servers equipped with 192 GB of RAM, 9 servers with 8 NVIDIA Tesla V100 cards each. The development worked on Prometheus in most of the project and recently the port to Ares HPC system was concluded, which is the cluster used for every recent result.

## 5.2. Execution profiling for the multiscale model

All the models are being developed in parallel with the orchestrator. Several versions of the models and the orchestrator were released during the development of the project. Countless additions and removals in the model implied changes in the topology applied and into the orchestrator.

The most computationally requiring part is the Agent-Based Model, although when run separately the model is fast, and additionally we have the possibility of using multiple GPUs, is necessary to run for hundreds or thousands of times per interaction. The team in charge of the ABM at the University of Sheffield (USFD), is the main one responsible for working on the speed-up of the PRIMAGE multiscale model and did several performance improvements simultaneously to the increments of the phenomena modelled. Table 5.1 shows the progress obtained since the most recent input file was defined. The input used was randomly selected. Although a reduction in memory consumption was reached, new increments made the model significantly slower from 13.8 to 13.9 versions. Which was improved in the up-to-date version and an extra gain was obtained using the V100 GPUs in Ares cluster.

Cluster	Prometheus	Prometheus	Prometheus	Ares
FGPU version	13.8.4	13.9.1	13.9.3	13.9.3
Walltime [s]	96	811	211	90
Memory consumption [Gb]	1.8	1.1	1.1	1.1

TABLE 5.1: Memory and time consumption gotten by USFD for the ABM model.

Moreover, the orchestrator underwent some rewriting to obtain a better performance. All the modules use the database to register their data. Initially, the particularisation and homogenisation modules were accessing the DB to insert or update each element individually. However, the bigger the problem becomes, the slower this approach got. In the prototype version only three variables were being homogenised, in the current state 24 variables are exchanged between the models and larger meshes started to be used as well, which implied in very slow relation modules, in particular, the homogenisation that was more dependent on the database. A complete rewrite of those functions was established to look for some performance optimisation for larger simulations. The number of queries was reduced, and they started to be executed by buckets of a pre-defined size allowing the same connection to be used for multiple insertions.

Table 5.2 compares a spherical tumour mesh of about 4 thousand elements using 10 GPUs and 1% of binning. The de-serialiser module is responsible for monitoring so the real-time taken to finish the model(s) impacts the number of verifications for the end of the model(s) making it impossible to disconnected it from the queueing time completely. A gain of 56% was obtained by switching to the Ares cluster.

Cluster	Prometheus	Prometheus	Ares
Orchestrator version	VPH-HF3 v2	VPH-FH3 v3	VPH-HP3 v3
Serialiser oxygen transport	0.3	0.6	2.4
Serialiser cell	0.1	0.1	0.8

Serialiser ABM	6.7	5.3	4.7
Serialiser structural	0.4	0.7	0.9
De-serialiser oxygen transport	13.7	14	11
De-serialiser cell	13.3	12	7.8
De-serialiser ABM	136.8	81	38.8
De-serialiser structural	14.6	3.7	8.1
Particularisation	194	127	130
Relation cell to ABM	5.4	5.3	5.3
Homogenisation	4215	143	136
Remeshing	96	97	97
<b>Total time</b>	<b>5476</b>	<b>1303</b>	<b>832</b>

TABLE 5.2: Real-time consumption of the orchestrator components of a 1-loop execution using an idealised spherical tumour of 4448 elements, 1% of binning and 10 GPUs. The ABM data presented represents the biggest time spent by one GPU.

From now on, we will use a 34 thousand elements mesh of a real tumour shape. In this case, using 8 cores for simulations of ANSYS model, the oxygen transport model took about 17.5 minutes to finish and the structural model 0.5 minutes. The ABM time consumption is proportional to the volume of the element, therefore the more refined is the mesh, the faster is the ABM. Thus, for a mesh of 34 thousand elements, the volume of an element (RVE) reduces dramatically, and the ABM run in an average of 6 seconds.

Aiming to assess the orchestrator performance in a shorter time frame we used a mock ABM model. Seven different runs were done only changing the size of the particularisation from 1% to 100% (no binning). Figure 5.1 presents the total execution time of each run using the mock ABM which run in less than one second. Naturally, the higher the sample used on particularisation the slower the multiscale model because more ABM executions are requested which implies in more inputs generated, larger DB tables to manage and more bins to homogenise. For lower values of particularisation size, there is a non-linearity in the time consumption behaviour once the models do not always take the same time to run, the queueing time is uncontrollable, and the delay of the database connection is also random.

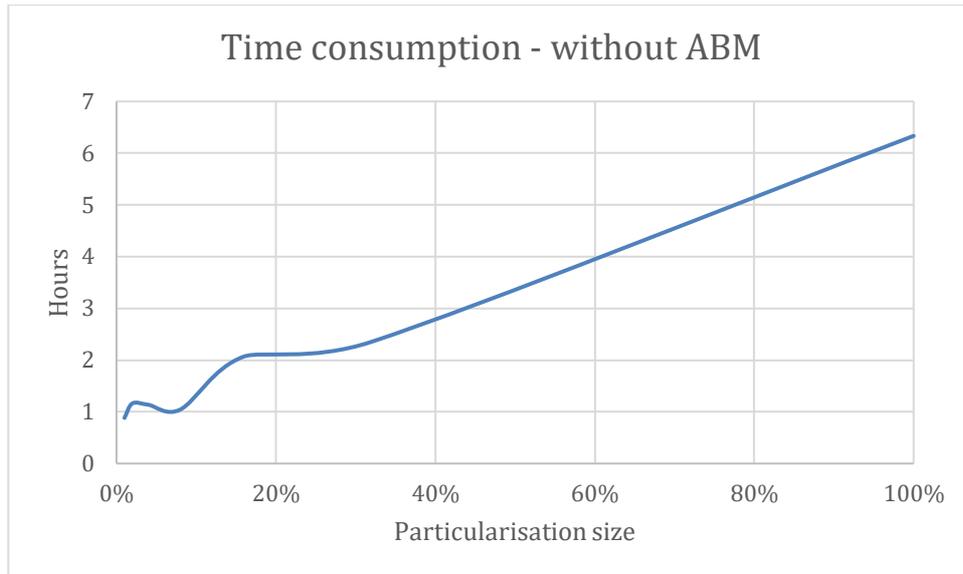


FIGURE 5.1: Total time consumption using a mock ABM varying the particularisation size using 10 GPUs on Ares cluster.

Figure 5.2 shows an estimation of the total execution time for a 1-loop simulation using the real ABM in 10 GPUs considering there is no queue time for the GPUs in the cluster. Although this is unrealistic as requiring multiple GPUs consecutively generates a penalty on using the cluster resources. A full resolution simulation (100% of particularisation) would take at least 12 hours in this case. Reaching a good homogenization for obtaining low errors for the 1% particularisation case, the time cost of 1-loop can be shorted 12 times or 8% of the full resolution case.

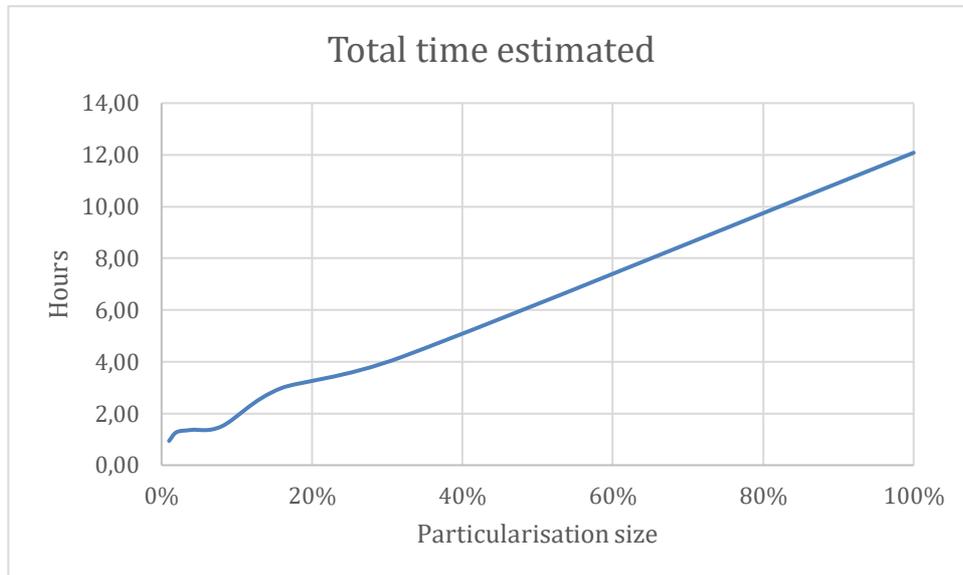


FIGURE 5.2: Estimation of the total time using the real ABM varying the particularisation size using 5 GPUs on Ares cluster

The following Tables analyses the time consumption of each component of the orchestrator for the same 34 thousand elements mesh in 1-loop. Table 5.3 presents the time expend for each instance of the serializer module. Serializing files is a relatively fast task but for running all the 34k elements this step can consume more than 1 hour, raising the necessity of doing particularisation and homogenisation.

Serialiser	Oxygen transport	Cell	ABM	Structural
1%	1.4	0.08	34.8	2.7
2%	2	0.13	164	2.4
4%	2	0.26	196	2
8%	1.4	0.12	213	1.6
16%	1.8	0.17	819	2.3
32%	1.6	0.15	1157	1.7
100%	1.8	0.16	3617	2.2

TABLE 5.3: Time consumption in seconds of each serialiser by particularisation size

Table 5.4 does the same for the de-serialiser module. It is important to highlight the de-serialiser module is also responsible for waiting the model to finish what causes more time consumption verifying the jobs' state, the de-serialisation by itself is not a computational-demand activity. Obviously, the ABM's de-serialiser is the most requiring instance once it needs to wait the most computational-expensive step of the orchestration and checking for multiple model simulations and multiple jobs to finish.

De-serialiser	Oxygen transport	Cell	ABM	Structural
1%	22.5	22.9	146	21.3
2%	81.9	71.5	371	76.5
4%	29.6	26.8	459	27.2
8%	25.7	25.8	522	24.6
16%	99.2	99.9	2011	102.6
32%	125	123	2929	117.6
100%	326	383	8997	154

TABLE 5.4: Time consumption in seconds of each de-serialiser by particularisation size.

Assessing the relation modules time consumption (table 5.5), the remeshing is equally fast regardless the particularisation and the improvements in the homogenisation removed its exponential behaviour of the previous versions. The sudden increment for 16%-case may be caused by database delays at the moment of the execution once the mathematical operations are the same for all cases and the number of queries is proportional to the mesh size.

The particularisation and the Relation 2 (between the cell model and the ABM) showed correlation to the particularisation degree. This happens in the particularisation because it uses several update queries which are slower once they lock the table before updating. The table 5.6 highlights how much of Relation time consumption is spent accessing the database. For the R4 instance this time is irrelevant because the database is only accessed at the beginning and the end of its execution. A opposite situation is found in the particularisation and homogenisation modules whose spend most of their activities in the database. That means their operations are not computer-expensive and an approach that does not register all the data in the database can be significantly faster. A smaller but similar situation is noticeable for the R2 instance, therefore strategies to speed-up these three modules must be more studied in future versions.

Relation	R1 - Particularisation	R2	R3 -Homogenisation	R4 - Remeshing
1%	950	70	786	63
2%	917	140	853	79
4%	1213	232	881	113
8%	1257	323	860	96
16%	1890	899	1485	107
32%	3718	1985	785	89
100%	8427	11043	950	99

TABLE 5.5: Time consumption in seconds of each relation module by particularisation size.

DB in relation	R1 - Particularisation	R2	R3 -Homogenisation	R4 - Remeshing
1%	99.9%	81.4%	98.2%	0.0%
2%	99.9%	80.0%	98.2%	0.0%
4%	99.8%	75.4%	96.0%	0.0%
8%	99.9%	73.4%	96.4%	0.0%
16%	99.9%	75.6%	84.7%	0.0%
32%	100.0%	81.2%	91.2%	0.0%
100%	100.0%	76.4%	98.1%	0.0%

TABLE 5.6: Database time consumption in each relation module by particularisation size.

During the development, it was noticed that the frequency of the polling can impact the total time consumption making the database slower. Then, a periodical polling was implemented for all the modules. New tests must be done to find an optimal period between polls, the table 5.7 shows polling for every three seconds has a negligible cost in the whole execution time.

Polling	Oxygen transport	Cell	ABM	Structural
1%	0.32%	0.34%	0.32%	0.32%
2%	0.31%	0.29%	0.33%	0.31%
4%	0.29%	0.31%	0.37%	0.33%
8%	0.28%	0.30%	0.36%	0.30%
16%	0.88%	0.85%	0.99%	0.68%
32%	0.35%	0.36%	0.56%	0.37%
100%	0.54%	0.45%	0.64%	0.43%

TABLE 5.7: Polling cost in an entire simulation by particularisation size.

### 5.3. Scalability and convergence of the particularisation algorithm

For better assessing the convergence of the particularisation, we need to obtain a solution without doing particularisation to have an expected solution to calculate the error caused by the homogenisation. The first setback we faced is that the ABM is stochastic, consequently, the multiscale model is stochastic as well. Aiming to make the ABM deterministic a mock model was built to use the pair of I/Os of the execution without particularisation to return as output always the same values for the same set of inputs, in the same way, it was done in the paper in chapter 4.

Doing so, we could repeat the analysis of the particularisation convergence to the updated version of the PRIMAGE multiscale model using a real tumour shape mesh. The particularisation size was varied from 1% to 32% and three different initial variables were used to do the particularisation: initial volume, oxygen concentration and initial cellularity. Figure 5.3 shows the Root Mean Square Error for all the cases. We can conclude that like in the spherical idealised tumour oxygen concentration is the most reliable variable found to do the particularisation although for 1% of particularisation level the error now is equal to 2.77% and 32% of particularisation is not enough to reduce the error significantly reaching error equal to 2.03%.

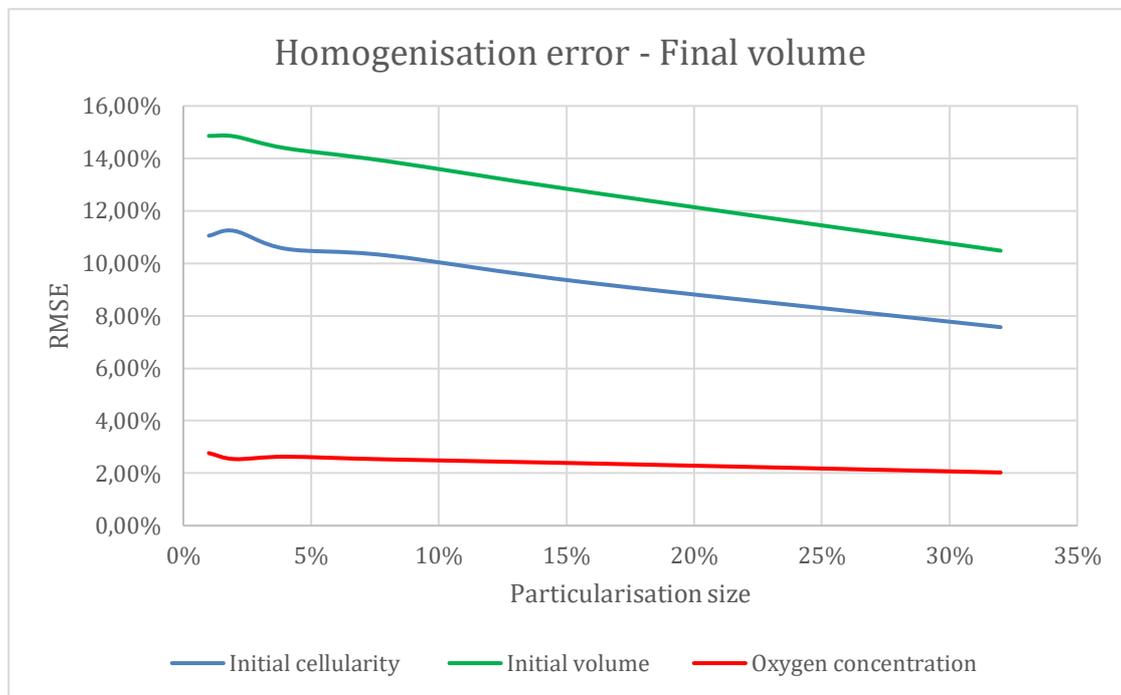


FIGURE 5.3: Comparison of the particularisation/homogenisation errors for different degrees of particularisation and making the particularisation based on initial volume, initial cellularity, and initial oxygen concentration.

To validate the algorithm an execution of a particularisation by oxygen of 100% size was done. The structural model is the model that gives the output of the orchestration. These outputs are composed of the volume of each element and the displacement of each node in each direction (x, y, and z). In figure 5.4 we can check the error for all structural outputs doing the particularisation by oxygen concentration. Apart from the displacement in the y direction, all variables have errors smaller than 3% for all particularisations tested. Tests with more interactions must be done to assess how these errors are impacting the tumour shape for the following interactions.

For reaching such a value for the y-displacement would be necessary to particularise using samples of more than 20%. Looking to the volume, to obtain an error smaller than 1%, like the error gotten in the idealised tumour, would be necessary a particularisation degree of about 67%. This would annulate most the advantage of doing particularisation.

These are the outputs generated by the homogenised input but how good is the interpolation of the ABM outputs? Figure 5.5 compares the average RMSE error for each of the variables that are being homogenised. For most of the variables is possible to see that homogenisation is very efficient in interpolating the values with low errors and is also evidence that increasing the particularisation size tends to improve the approximation. Volume, oxygen concentration and the cellularities of alive cells are considered the most critical variables because they cause more impact in the following steps. The worst case is the volume, further analysis must be done to confirm if this is what is related to the high displacement errors in the final outputs.

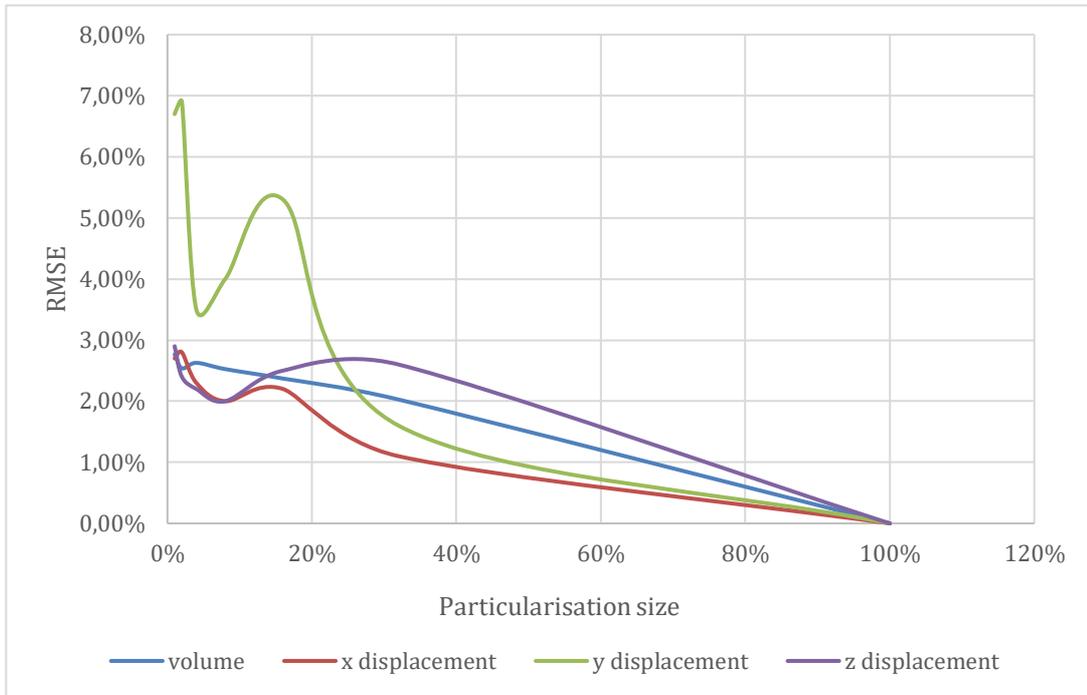
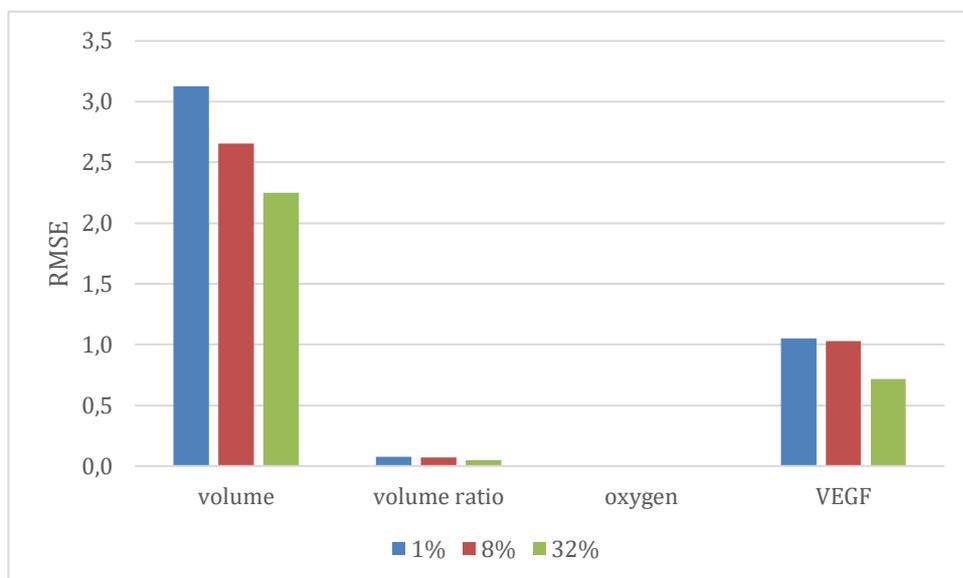


FIGURE 5.4: Homogenisation error for structural model outputs particularising by oxygen concentration



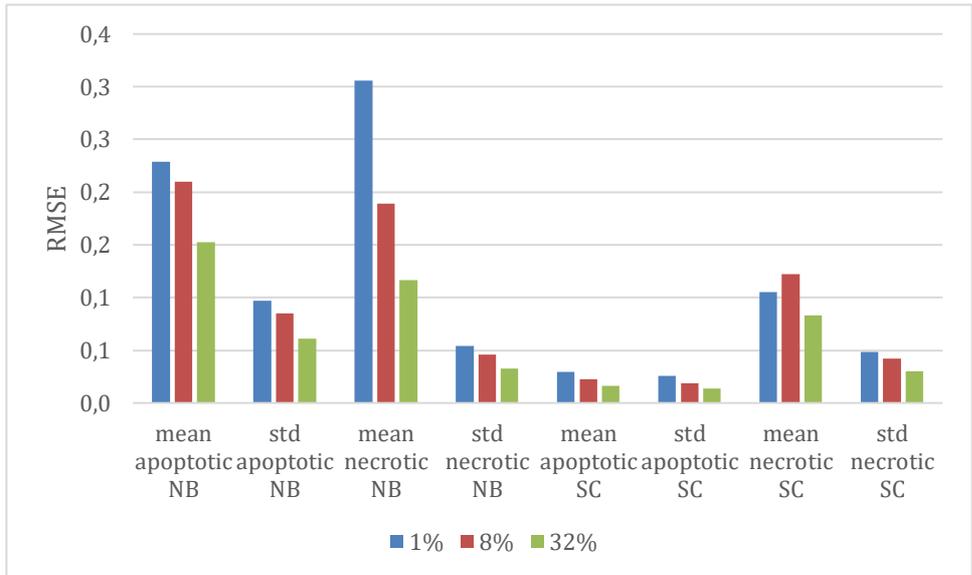
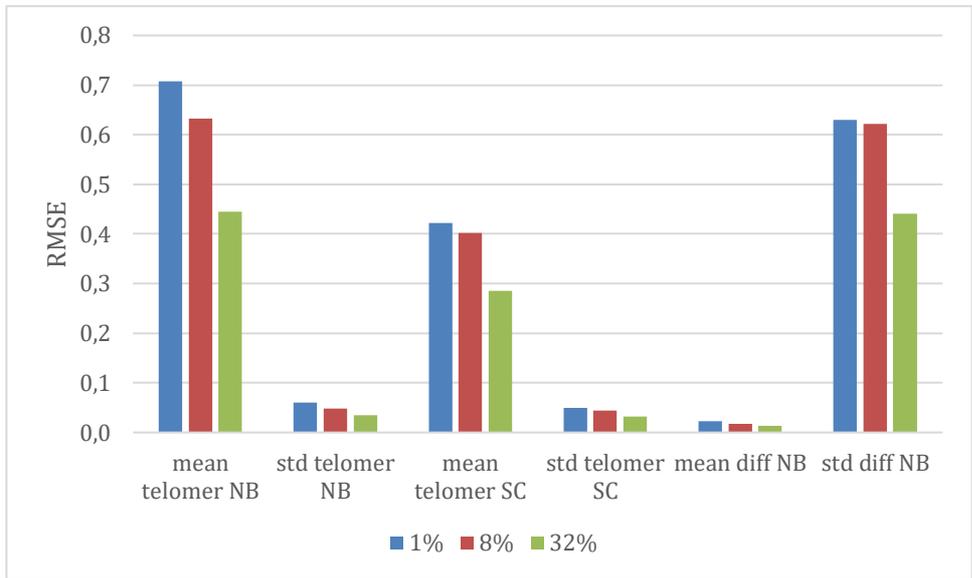
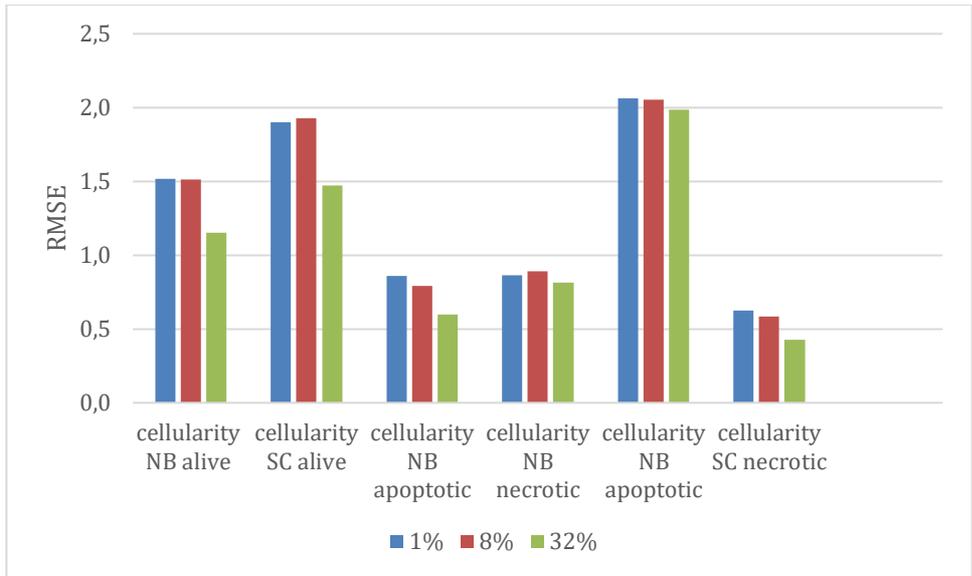


FIGURE 5.5: The average error of the homogenised values for three executions doing the particularisation by oxygen. Each execution was done with 1%, 8% and 32% of particularisation respectively.

The errors are considered small for most of variables although further tests may be done to access how those errors are propagated for long simulations. In multiple loops simulations, this analysis is impossible. The remeshing will change the mesh for all the loops after the initial interaction, based on the outputs generated by the first homogenisation. Also, the ABM is stochastic, we cannot use only the outputs of 1-loop full execution (no particularisation) to build a mock model and, even so, is very time-requiring to run a whole chemotherapy without particularisation.

A different approach must be applied for evaluating the scalability of the particularisation algorithm. This study is initialised in section 5.4, however, the models are still not well calibrated. Calibration of multiscale models is complex and is out of the scope of this work. The next section present preliminary analysis of the particularisation / homogenisation errors for the whole chemotherapy simulations.

#### 5.4. Full simulation results

In the current state of the models, the PRIMAGE multiscale model is not well calibrated. The tumour growth, the tumour shape and most of the variables are not validated. Given the large number of parameters and the time required to run a full simulation, the calibration of multiscale models is complex and slow. However, this does not prevent the analysis of the particularization error. This section aims to assess it for 6 loops (12 weeks of treatment) simulations that represent the whole chemotherapy of the case.

As explained before, it is impossible to do the same analysis done for one-loop simulations because after each loop is done a remeshing. So, even forcing determinism in the ABM the error caused by the homogenisation will imply in different meshes after the remeshing and from this point on, all the solutions will not be comparable to the full-resolution simulation. Those simulations are also very time-requiring what makes the full-resolution simulation unfeasible. Also, the stochasticity of the ABM, in theory, would request multiples simulation with the same parameters.

Aiming to reduce the number of simulations and the duration of them. Four simulations were done in order to assess the asymptotic behavior of the three most critical variables at the end of a loop (after the remeshing to the following loop): volume, oxygen concentration and cellularity. As we are comparing values between different meshes, it was considered for each case the total volume obtained and the average of the oxygen concentration and the cellularity.

Figure 5.6 compares these values for four different simulations for each of the 6 loops simulated, changing only the size of the particularization. Observing the total volume behavior, the solution with a smaller particularisation presented values not aligned with the other simulations. This is an indicative that 4% of particularisation may not be enough to reach reliable solutions although more executions would be necessary to ensure this difference was not caused by the stochasticity of the ABM

Analysing the average oxygen concentration, the homogenisation seems to make a good approximation for all the cases tested because the overall tendence and values are similar for all of them. Slight differences are expected since the PRIMAGE model is not deterministic. For the cellularity values, the 4%-sample particularisation again unattached from the other solutions, Further simulations must be performed to ensure this was not caused by stochasticity. This analysis evidenced that particularisation between 8% to 20% does not bring significant differences in the final solution,

which induces that 8% of particularisation may present neglectable error in comparison to a full-resolution model's solution.

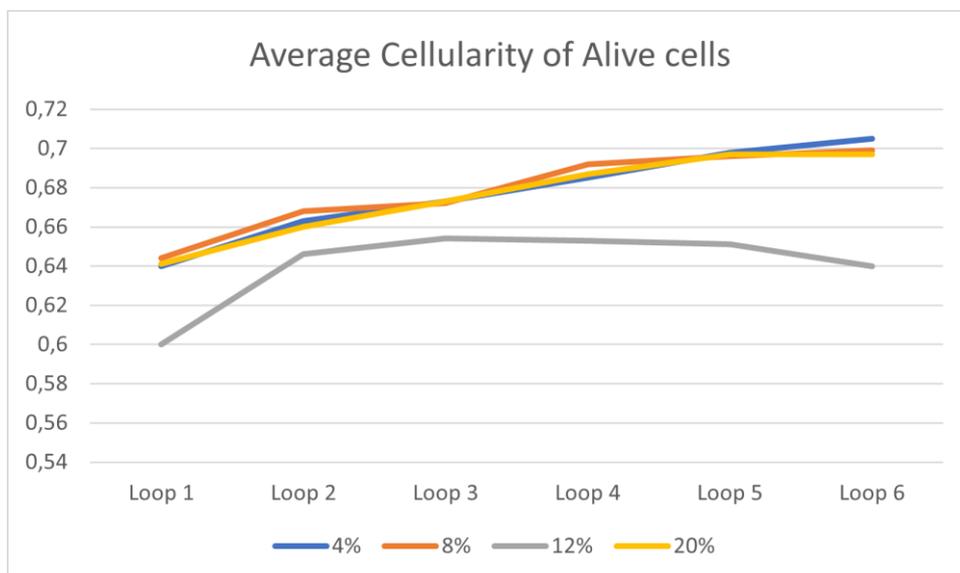
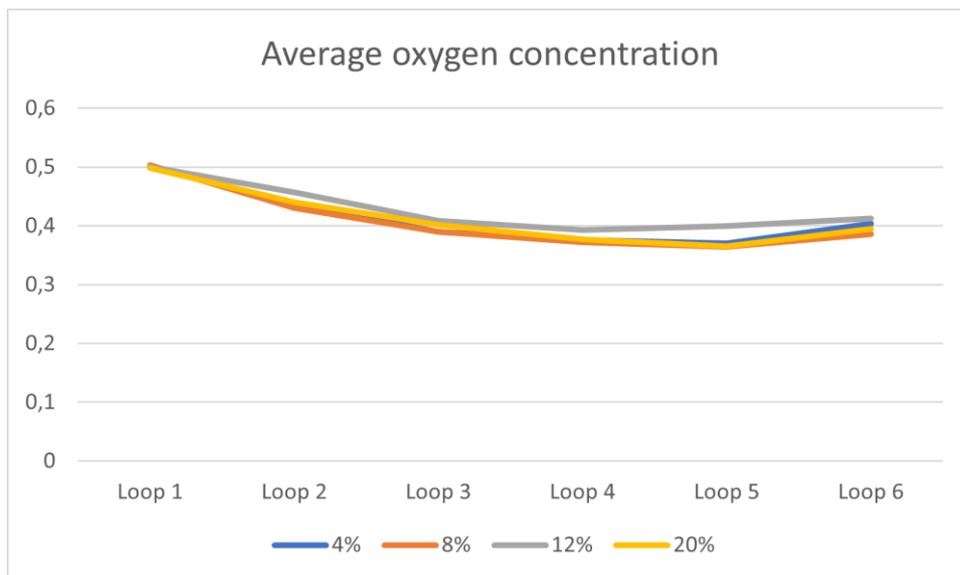
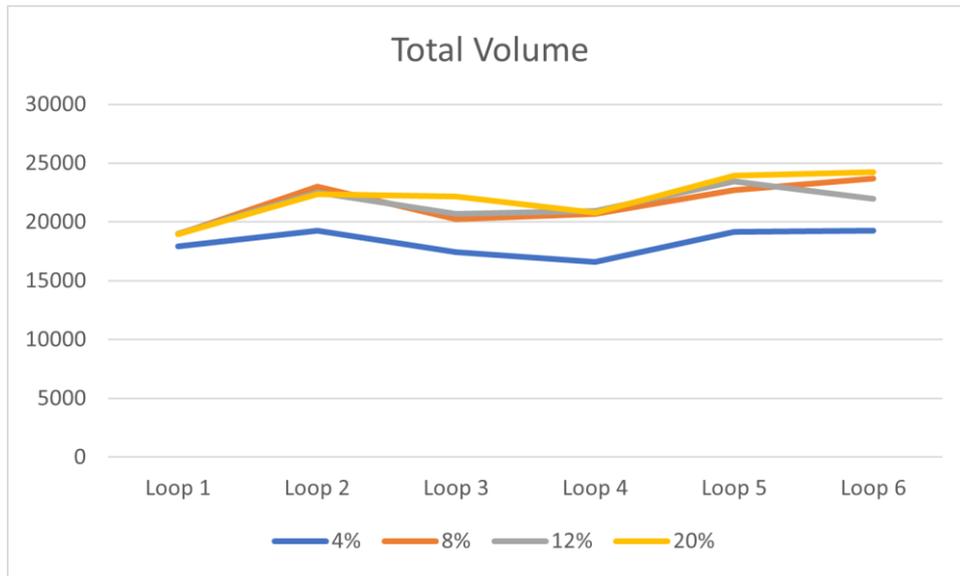


FIGURE 5.6: Values of volume, oxygen concentration and cellularity of alive cells for a 6 loops simulation. Four simulations were performed only changing the particularisation level to 4%, 8%, 12% and 20%.

The Figure 5.7 shows the wall time for each of these simulations. Even though a full-resolution simulation is possible, its results are not very relevant because the multiscale model is not deterministic. The real time execution can be shortened using multiple GPUs, although it implies in longer queueing time. The best number of GPUs must be defined for the end application in the PRIMAGE platform.

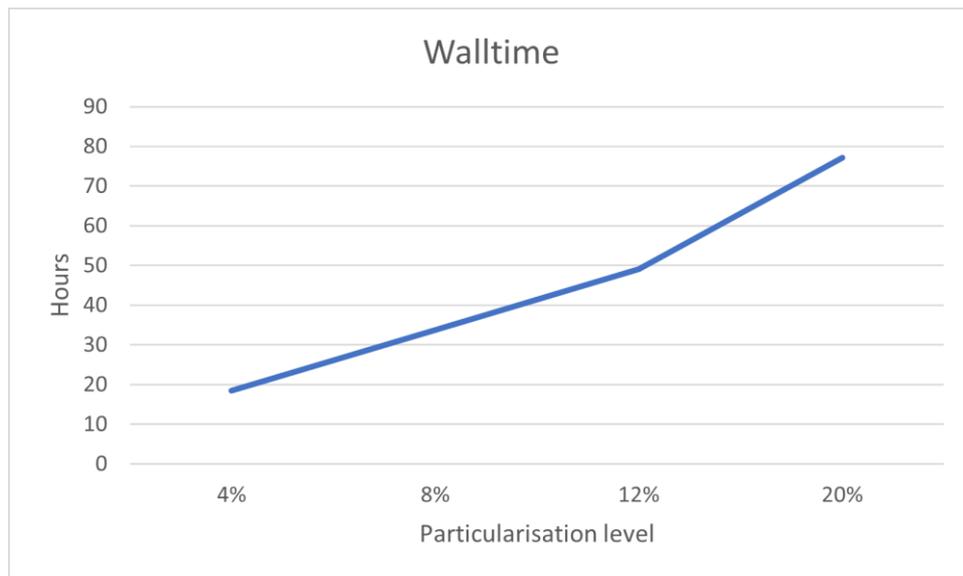


FIGURE 5.7: Wall time for 6 loop simulations. Four simulations were performed only changing the particularisation level to 4%, 8%, 12% and 20%.

This analysis must be re-done after the model's calibration, because changing the model's parameters will impact the rules that generate their outputs and consequently, will impact the multiscale model.

## 5.5. Conclusions and future works

It was developed a framework for orchestrating several models in complex architectures which aimed to be as simple as possible, focused on a reusable template and modularized to be easier to maintain. The tool was applied for a complex oncology multiscale model in PRIMAGE project which were developed in parallel with the single-scale models addressing all the demand requested. The framework applied to the neuroblastoma multiscale model will be connected to the PRIMAGE platform to be used for real cases and is being used to support the modelers on tweaking the models.

Naturally there are more to be done, the same idea was applied in a smaller problem (UISS-TB project) in the early beginning of this project which guided the current architecture, but the tool still needs to be applied to other problems aiming to better work on its reusability.

Different approaches in the database architecture can be assessed too. Such as using NoSQL database in order to make the database more generic and easier adaptable for different problems. The profiling showcased some progress in the computational cost of the orchestrator, but more work must be done. The excessive dependence of the database is implying in slower relation modules and the period of the polling should be better assessed for each case.

The framework also included a binning strategy of particularization and homogenization that although showed useful for reducing efficiently the time consumption and, its accuracy was very efficient for an idealized tumour, the same did not happen to a more complex multiscale model with a complex shape and tens of variables. New approaches should be tested for example do not use linear interpolation in the homogenisation or even enable the possibility of applying a different interpolation strategy for each variable homogenized. Adaptive methods are highly used in the literature and could be tried as well.

## 6. References

- Alemani, D., Pappalardo, F., Pennisi, M., Motta, S., Brusici, V., 2012. Combining cellular automata and lattice Boltzmann method to model multiscale avascular tumor growth coupled with nutrient diffusion and immune competition. *Journal of Immunological Methods* 376, 55–68. <https://doi.org/10.1016/j.jim.2011.11.009>
- Alowayyed, S., Groen, D., Coveney, P.V., Hoekstra, A.G., 2017. Multiscale computing in the exascale era. *Journal of Computational Science* 22, 15–25. <https://doi.org/10.1016/j.jocs.2017.07.004>
- American Cancer Society, A., 2022. *Cancer Facts & Figures 2022* 80.
- American Cancer Society, A., 2021a. Tests For Neuroblastoma | Neuroblastoma Ultrasound [WWW Document]. URL <https://www.cancer.org/cancer/neuroblastoma/detection-diagnosis-staging/how-diagnosed.html> (accessed 11.2.22).
- American Cancer Society, A., 2021b. Neuroblastoma Surgery | Treating Neuroblastoma [WWW Document]. URL <https://www.cancer.org/cancer/neuroblastoma/treating/surgery.html> (accessed 11.2.22).
- American Childhood Cancer Organization, A., 2018. Neuroblastoma Childhood Cancer: Signs and Symptoms. ACCO. URL <https://www.acco.org/blog/neuroblastoma-childhood-cancer-signs-and-symptoms/> (accessed 11.2.22).
- Associazione Italiana di Oncologia Medica, A., Associazione Italiana Registri Tumor, A., Società Italiana di Anatomia Patologica e di Citopatologia, S., 2020. I numeri del cancro in Italia.

- Associazione Italiana per la Ricerca sul Cancro, A., 2019. Neuroblastoma bambini: sintomi, diagnosi e cure [WWW Document]. URL <https://www.airc.it/cancro/informazioni-tumori/guida-ai-tumori-pediatrici/neuroblastoma-pediatrico> (accessed 11.2.22).
- Babuška, I., Motamed, M., Tempone, R., 2014. A stochastic multiscale method for the elastodynamic wave equation arising from fiber composites. *Computer Methods in Applied Mechanics and Engineering* 276, 190–211. <https://doi.org/10.1016/j.cma.2014.02.018>
- Bayeva, N., Coll, E., Piskareva, O., 2021. Differentiating Neuroblastoma: A Systematic Review of the Retinoic Acid, Its Derivatives, and Synergistic Interactions. *J Pers Med* 11, 211. <https://doi.org/10.3390/jpm11030211>
- Beaumont, H., Souchet, S., Labatte, J.M., Iannessi, A., Tolcher, A.W., 2015. Changes of lung tumour volume on CT - prediction of the reliability of assessments. *Cancer Imaging* 15, 17. <https://doi.org/10.1186/s40644-015-0052-2>
- Bhattacharya, P., Li, Q., Lacroix, D., Kadiramanathan, V., Viceconti, M., 2021. A systematic approach to the scale separation problem in the development of multiscale models. *PLoS One* 16, e0251297. <https://doi.org/10.1371/journal.pone.0251297>
- Borgdorff, J., Mamonski, M., Bosak, B., Kurowski, K., Ben Belgacem, M., Chopard, B., Groen, D., Coveney, P.V., Hoekstra, A.G., 2014. Distributed multiscale computing with MUSCLE 2, the Multiscale Coupling Library and Environment. *Journal of Computational Science* 5, 719–731. <https://doi.org/10.1016/j.jocs.2014.04.004>
- Canadian Cancer Society, A., 2016. How cancer starts, grows and spreads [WWW Document]. URL <https://cancer.ca/en/cancer-information/what-is-cancer/how-cancer-starts-grows-and-spreads> (accessed 11.2.22).
- Canadian Cancer Society, C., n.d. How cancer starts, grows and spreads [WWW Document]. URL <https://cancer.ca/en/cancer-information/what-is-cancer/how-cancer-starts-grows-and-spreads> (accessed 11.2.22).
- Cancer Research UK, C., 2014. The immune system and cancer [WWW Document]. Cancer Research UK. URL <https://www.cancerresearchuk.org/about-cancer/what-is-cancer/body-systems-and-cancer/the-immune-system-and-cancer> (accessed 11.2.22).
- Cancer.net, A., 2012. Neuroblastoma - Childhood - Types of Treatment [WWW Document]. Cancer.Net. URL <https://www.cancer.net/cancer-types/neuroblastoma-childhood/types-treatment> (accessed 11.2.22).
- Celis, J.S., Wibberg, D., Ramírez-Portilla, C., Rupp, O., Sczyrba, A., Winkler, A., Kalinowski, J., Wilke, T., 2018. Binning enables efficient host genome reconstruction in cnidarian holobionts. *GigaScience* 7, giy075. <https://doi.org/10.1093/gigascience/gyi075>
- Children’s Cancer and Leukemia Group, C., 2019. Outcome after high-risk neuroblastoma treatment in Europe [WWW Document]. URL <https://www.cclg.org.uk/keeping-you-informed/outcome-after-high-risk-neuroblastoma-treatment-in-europe> (accessed 11.2.22).
- Collis, J., Hubbard, M.E., O’dea, R.D., 2017. A multi-scale analysis of drug transport and response for a multi-phase tumour model. *European Journal of Applied Mathematics* 28, 499–534. <https://doi.org/10.1017/S0956792516000413>
- Dahmann, J.S., Fujimoto, R.M., Weatherly, R.M., 1998. The DoD High Level Architecture: an update, in: 1998 Winter Simulation Conference. Proceedings (Cat. No.98CH36274). Presented at the IEEE Winter Simulation Conference, IEEE, Washington, DC, USA, pp. 797–804. <https://doi.org/10.1109/WSC.1998.745066>
- de la Cruz, R., Guerrero, P., Calvo, J., Alarcón, T., 2017. Coarse-graining and hybrid methods for efficient simulation of stochastic multi-scale models of tumour growth. *J Comput Phys* 350, 974–991. <https://doi.org/10.1016/j.jcp.2017.09.019>

- de Melo Quintela, B., Hervás-Raluy, S., García-Aznar, J.M., Walker, D., Wertheim, K.Y., Viceconti, M., 2022. A theoretical analysis of the scale separation in a model to predict solid tumour growth. *J Theor Biol* 547, 111173. <https://doi.org/10.1016/j.jtbi.2022.111173>
- Engquist, B., Souganidis, P.E., 2008. Asymptotic and numerical homogenization. *Acta Numerica* 17, 147–190. <https://doi.org/10.1017/S0962492906360011>
- Falcone, J.-L., Chopard, B., Hoekstra, A., 2010. MML: towards a Multiscale Modeling Language. *Procedia Computer Science, ICCS 2010* 1, 819–826. <https://doi.org/10.1016/j.procs.2010.04.089>
- Gostoli, S., Montecchiarini, M., Urgese, A., Ferrara, F., Polifemo, A.M., Ceroni, L., Gasparri, A., Rafanelli, C., Cennamo, V., 2021. The clinical utility of a comprehensive psychosomatic assessment in the program for colorectal cancer prevention: a cross-sectional study. *Sci Rep* 11, 15575. <https://doi.org/10.1038/s41598-021-95171-8>
- Granata, C., Fagnani, A.M., Gambini, C., Boglino, C., Bagnulo, S., Cecchetto, G., Federici, S., Inserra, A., Michelazzi, A., Riccipettoni, G., Rizzo, A., Tamaro, P., Jasonni, V., Bernardi, B.D., 2000. Features and outcome of neuroblastoma detected before birth. *Journal of Pediatric Surgery* 35, 88–91. [https://doi.org/10.1016/S0022-3468\(00\)80020-2](https://doi.org/10.1016/S0022-3468(00)80020-2)
- Groen, D., Bhati, A.P., Suter, J., Hetherington, J., Zasada, S.J., Coveney, P.V., 2016. FabSim: Facilitating computational research through automation on large-scale and distributed e-infrastructures. *Computer Physics Communications* 207, 375–385. <https://doi.org/10.1016/j.cpc.2016.05.020>
- Groen, D., Knap, J., Neumann, P., Suleimenova, D., Veen, L., Leiter, K., 2019. Mastering the scales: a survey on the benefits of multiscale computing software. *Phil. Trans. R. Soc. A* 377, 20180147. <https://doi.org/10.1098/rsta.2018.0147>
- Hanahan, D., Weinberg, R.A., 2011. Hallmarks of Cancer: The Next Generation. *Cell* 144, 646–674. <https://doi.org/10.1016/j.cell.2011.02.013>
- Hanahan, D., Weinberg, R.A., 2000. The Hallmarks of Cancer. *Cell* 100, 57–70. [https://doi.org/10.1016/S0092-8674\(00\)81683-9](https://doi.org/10.1016/S0092-8674(00)81683-9)
- He, Y., Kodali, A., Wallace, D.I., 2018. Predictive Modeling of Neuroblastoma Growth Dynamics in Xenograft Model After Bevacizumab Anti-VEGF Therapy. *Bull Math Biol* 80, 2026–2048. <https://doi.org/10.1007/s11538-018-0441-3>
- Hitzer, B., León-Sanz, P., 2016. The Feeling Body and Its Diseases: How Cancer Went Psychosomatic in Twentieth-Century Germany. *Osiris* 31, 67–93. <https://doi.org/10.1086/687591>
- Hung, G.-Y., Horng, J.-L., Yen, H.-J., Lee, C.-Y., 2015. Infant Cancer in Taiwan: Incidence and Trends (1995-2009). *PLOS ONE* 10, e0130444. <https://doi.org/10.1371/journal.pone.0130444>
- Huq, F., Brannon, R., Graham-Brady, L., 2016. An efficient binning scheme with application to statistical crack mechanics. *International Journal for Numerical Methods in Engineering* 105, 33–62. <https://doi.org/10.1002/nme.4959>
- Kasemeier-Kulesa, J.C., Schnell, S., Woolley, T., Spengler, J.A., Morrison, J.A., McKinney, M.C., Pushel, I., Wolfe, L.A., Kulesa, P.M., 2018. Predicting neuroblastoma using developmental signals and a logic-based model. *Biophys Chem* 238, 30–38. <https://doi.org/10.1016/j.bpc.2018.04.004>
- Knap, J., Spear, C., Leiter, K., Becker, R., Powell, D., 2016. A computational framework for scale-bridging in multi-scale simulations: A COMPUTATIONAL FRAMEWORK FOR SCALE-BRIDGING. *Int. J. Numer. Meth. Engng* 108, 1649–1666. <https://doi.org/10.1002/nme.5270>
- Kojic, M., Milosevic, M., Kojic, N., Kim, K., Ferrari, M., Ziemys, A., 2014. A multiscale MD–FE model of diffusion in composite media with internal surface interaction based on numerical homogenization procedure. *Computer Methods in Applied Mechanics and Engineering* 269, 123–138. <https://doi.org/10.1016/j.cma.2013.11.010>

- Leiter, K.W., Barnes, B.C., Becker, R., Knap, J., 2018. Accelerated scale-bridging through adaptive surrogate model evaluation. *Journal of Computational Science* 27, 91–106. <https://doi.org/10.1016/j.jocs.2018.04.010>
- Martí-Bonmatí, L., Alberich-Bayarri, Á., Ladenstein, R., Blanquer, I., Segrelles, J.D., Cerdá-Alberich, L., Gkontra, P., Hero, B., García-Aznar, J.M., Keim, D., Jentner, W., Seymour, K., Jiménez-Pastor, A., González-Valverde, I., Martínez de las Heras, B., Essiaf, S., Walker, D., Rochette, M., Bubak, M., Mestres, J., Viceconti, M., Martí-Besa, G., Cañete, A., Richmond, P., Wertheim, K.Y., Gubala, T., Kasztelnik, M., Meizner, J., Nowakowski, P., Gilpérez, S., Suárez, A., Aznar, M., Restante, G., Neri, E., 2020. PRIMAGE project: predictive in silico multiscale analytics to support childhood cancer personalised evaluation empowered by imaging biomarkers. *European Radiology Experimental* 4, 22. <https://doi.org/10.1186/s41747-020-00150-9>
- Matthay, K.K., Maris, J.M., Schleiermacher, G., Nakagawara, A., Mackall, C.L., Diller, L., Weiss, W.A., 2016. Neuroblastoma. *Nat Rev Dis Primers* 2, 16078. <https://doi.org/10.1038/nrdp.2016.78>
- Merla, C., Liberti, M., Consales, C., Denzi, A., Apollonio, F., Marino, C., Benassi, B., 2019. Evidences of plasma membrane-mediated ROS generation upon ELF exposure in neuroblastoma cells supported by a computational multiscale approach. *Biochimica et Biophysica Acta (BBA) - Biomembranes* 1861, 1446–1457. <https://doi.org/10.1016/j.bbamem.2019.06.005>
- National Institute of Cancer, N., 2021. What Is Cancer? [WWW Document]. URL <https://www.cancer.gov/about-cancer/understanding/what-is-cancer> (accessed 11.2.22).
- Oncoguia, 2020. Câncer infantojuvenil: mais de 8 mil novos casos por ano - Instituto Oncoguia [WWW Document]. URL <http://www.oncoguia.org.br/conteudo/cancer-infantojuvenil-mais-de-8-mil-novos-casos-por-ano/15220/7/> (accessed 11.2.22).
- O'Neill, P., Connolly, R., Connolly, M., Soon, W., Chimani, B., Crok, M., de Vos, R., Harde, H., Kajaba, P., Nojarov, P., Przybylak, R., Rasol, D., Skrynyk, Oleg, Skrynyk, Olesya, Štěpánek, P., Wypych, A., Zahradníček, P., 2022. Evaluation of the Homogenization Adjustments Applied to European Temperature Records in the Global Historical Climatology Network Dataset. *Atmosphere* 13, 285. <https://doi.org/10.3390/atmos13020285>
- Organização Pan-Americana da Saúde, O., 2020. Câncer [WWW Document]. URL <https://www.paho.org/pt/topicos/cancer> (accessed 11.2.22).
- Paladim, D.A., Moitinho de Almeida, J.P., Bordas, S.P.A., Kerfriden, P., 2017. Guaranteed error bounds in homogenisation: an optimum stochastic approach to preserve the numerical separation of scales. *International Journal for Numerical Methods in Engineering* 110, 103–132. <https://doi.org/10.1002/nme.5348>
- Pizzi, G., Cepellotti, A., Sabatini, R., Marzari, N., Kozinsky, B., 2016. AiiDA: automated interactive infrastructure and database for computational science. *Computational Materials Science* 111, 218–230. <https://doi.org/10.1016/j.commatsci.2015.09.013>
- Raffaghello, L., Prigione, I., Airoidi, I., Camoriano, M., Morandi, F., Bocca, P., Gambini, C., Ferrone, S., Pistoia, V., 2005. Mechanisms of immune evasion of human neuroblastoma. *Cancer Letters, Advances in basic and translational neuroblastoma research* 228, 155–161. <https://doi.org/10.1016/j.canlet.2004.11.064>
- Richmond, P., Chisholm, R., Heywood, P., Leach, M., Kabiri Chimeh, M., 2021. FLAME GPU. <https://doi.org/10.5281/zenodo.5769677>
- Richmond, P., Walker, D., Coakley, S., Romano, D., 2010. High performance cellular level agent-based simulation with FLAME for the GPU. *Briefings in Bioinformatics* 11, 334–347. <https://doi.org/10.1093/bib/bbp073>
- Ritchie, H., Spooner, F., Roser, M., 2018. Causes of death. *Our World in Data*.

- Santos, M. de O., 2019. Incidência, Mortalidade e Morbidade Hospitalar por Câncer em Crianças, Adolescentes e Adultos Jovens no Brasil: Informações dos Registros de Câncer e do Sistema de Mortalidade. *Rev. Brasileira.De.Cancerologia* 64, 439–440. <https://doi.org/10.32635/2176-9745.RBC.2018v64n3.56>
- Siegel, R.L., Miller, K.D., Fuchs, H.E., Jemal, A., 2021. Cancer Statistics, 2021. *CA: A Cancer Journal for Clinicians* 71, 7–33. <https://doi.org/10.3322/caac.21654>
- Spix, C., Pastore, G., Sankila, R., Stiller, C.A., Steliarova-Foucher, E., 2006. Neuroblastoma incidence and survival in European children (1978–1997): Report from the Automated Childhood Cancer Information System project. *European Journal of Cancer* 42, 2081–2091. <https://doi.org/10.1016/j.ejca.2006.05.008>
- Stamatakos, G., Dionysiou, D., Misichroni, F., Graf, N., van Gool, S., Bohle, R., Dong, F., Viceconti, M., Marias, K., Sakkalis, V., Forgo, N., Radhakrishnan, R., Byrne, H., Guiot, C., Buechler, P., Neri, E., Bucur, A., de Bono, B., Testi, D., Tsiknakis, M., 2014. Computational Horizons In Cancer (CHIC): Developing Meta- and Hyper-Multiscale Models and Repositories for In Silico Oncology – a Brief Technical Outline of the Project. *Proc 2014 6th Int Adv Res Workshop In Silico Oncol Cancer Investig* (2014) 2014, 10.1109/iarwisoci.2014.7034630. <https://doi.org/10.1109/iarwisoci.2014.7034630>
- Stamatakos, G.S., Giatili, S.G., 2017. A Numerical Handling of the Boundary Conditions Imposed by the Skull on an Inhomogeneous Diffusion-Reaction Model of Glioblastoma Invasion Into the Brain: Clinical Validation Aspects. *Cancer Inform* 16, 1176935116684824. <https://doi.org/10.1177/1176935116684824>
- Tas, M.L., Reedijk, A.M.J., Karim-Kos, H.E., Kremer, L.C.M., Ven, C.P. van de, Dierselhuis, M.P., Eijkelenburg, N.K.A. van, Grotel, M. van, Kraal, K.C.J.M., Peek, A.M.L., Coebergh, J.W.W., Janssens, G.O.R., Keizer, B. de, Krijger, R.R. de, Pieters, R., Tytgat, G. a. M., Noesel, M.M. van, 2020. Neuroblastoma between 1990 and 2014 in the Netherlands: Increased incidence and improved survival of high-risk neuroblastoma. *European Journal of Cancer* 124, 47–55. <https://doi.org/10.1016/j.ejca.2019.09.025>
- Vernerey, F.J., Kabiri, M., 2014. Adaptive concurrent multiscale model for fracture and crack propagation in heterogeneous media. *Computer Methods in Applied Mechanics and Engineering* 276, 566–588. <https://doi.org/10.1016/j.cma.2014.03.004>
- Viceconti, M., 2012. Image-based, personalised and multiscale modelling to predict the risk of fracture in osteoporotic patients: the VPHOP integrated project 4657 words. <https://doi.org/10.1594/ECR2012/C-0830>
- Viceconti, M., Bnà, S., Tartarini, D., Sfakianakis, S., Grogan, J., Walker, D., Gamble, S., Testi, D., 2018. VPH-HF: A software framework for the execution of complex subject-specific physiology modelling workflows. *Journal of Computational Science* 25, 101–114. <https://doi.org/10.1016/j.jocs.2018.02.009>
- Ward, Z.J., Yeh, J.M., Bhakta, N., Frazier, A.L., Girardi, F., Atun, R., 2019. Global childhood cancer survival estimates and priority-setting: a simulation-based analysis. *The Lancet Oncology* 20, 972–983. [https://doi.org/10.1016/S1470-2045\(19\)30273-6](https://doi.org/10.1016/S1470-2045(19)30273-6)
- Wienke, J., Dierselhuis, M.P., Tytgat, G.A.M., Künkele, A., Nierkens, S., Molenaar, J.J., 2021. The immune landscape of neuroblastoma: Challenges and opportunities for novel therapeutic strategies in pediatric oncology. *European Journal of Cancer* 144, 123–150. <https://doi.org/10.1016/j.ejca.2020.11.014>
- Wolstencroft, K., Haines, R., Fellows, D., Williams, A., Withers, D., Owen, S., Soiland-Reyes, S., Dunlop, I., Nenadic, A., Fisher, P., Bhagat, J., Belhajjame, K., Bacall, F., Hardisty, A., Nieva de la Hidalga, A., Balcazar Vargas, M.P., Sufi, S., Goble, C., 2013. The Taverna workflow suite: designing

and executing workflows of Web Services on the desktop, web or in the cloud. *Nucleic Acids Res* 41, W557–W561. <https://doi.org/10.1093/nar/gkt328>

World Cancer Research Fund International, W., 2022. Global cancer data by country. WCRF International. URL <https://www.wcrf.org/cancer-trends/global-cancer-data-by-country/> (accessed 11.2.22).

World Health Organization, 2021a. Cancer today [WWW Document]. URL <http://gco.iarc.fr/today/home> (accessed 11.2.22).

World Health Organization, 2021b. CureAll framework: WHO global initiative for childhood cancer: increasing access, advancing quality, saving lives. World Health Organization, Geneva.

World Health Organization, 2020a. Assessing national capacity for the prevention and control of noncommunicable diseases: report of the 2019 global survey. World Health Organization, Geneva.

World Health Organization, 2020b. WHO report on cancer: setting priorities, investing wisely and providing care for all. World Health Organization, Geneva.

Zander, E., 1983. [Cancer--a psychosomatic disease?]. *Z Psychosom Med Psychoanal* 29, 363–379.

## APPENDIX 1: Inputs and Outputs of PRIMAGE models

Image processor (UNIZAR)	
Input files	Description
DCE MR Images	Files of Diffusion-weighted magnetic resonance imaging

DWI MR Images	Files of dynamic contrast enhanced T1 MRI images
Geometry	File describing the surface from GIBI analysis
<b>Output files</b>	<b>Description</b>
i_cellularity.txt	Initial cellularity
i_vascularization.txt	Initial vascularization
ic_oxygen.txt	Initial oxygen concentration
mesh_nodes.inp	Nodes of the mesh
mesh_elements.inp	Elements of the mesh
<b>Oxygen Transport (UNIZAR)</b>	
<b>Input files</b>	<b>Description</b>
mesh_nodes.inp	Nodes of the mesh
mesh_elements.inp	Elements of the mesh
ic_oxygen.txt	initial oxygen concentration
i_vascularization.txt	Values of Ktrans
<b>Output files</b>	<b>Description</b>
oxygen_i.txt	updated oxygen concentration
volume_i.txt	volume of each element
<b>Mechanical (UNIZAR)</b>	
<b>Input file</b>	<b>Description</b>
mesh_nodes.inp	Nodes of the mesh
mesh_structural.inp	Elements of the mesh (doubled)
materials_elastic.inp	Material properties defined in each element
growth_APDL.txt	volume increment for each element (in temperature terms)
<b>Output files</b>	<b>Description</b>
geometry.stl	Tumor geometry
volume_f.txt	Volume after tumor growth
<b>ABM (USFD)</b>	
<b>Inputs</b>	<b>Description</b>
volume	Volume of the element
cellularity	Relative volume occupied by alive, apoptotic and necrotic cells
telomer length	Mean and std length on NB and SC
o2	Oxygen concentration
death signal	Mean and std of apoptotic and necrotic signals for NB and SC
MYCN_amp	MYCN amplification
TERT_rargmt	TERT rearrangement
ATRX_inact	TATRX inactivation

ALT	ALT status
ALK	ALK activating mutation or amplification
MKI	MKI cataloged into 3 classes
MKI confidence score	Confidence Score of MKI values
Diff. grade	Differentiation grades cataloged into 3 classes
Diff. grade confidence score	Confidence Score of Differentiation Grade values
effect_protein.txt	Probability of the drug effect on tumor areas
<b>Output</b>	<b>Description</b>
volume	Volume of the element
cellularity	Relative volume occupied by alive, apoptotic and necrotic cells
telomer length	Mean and std length on NB and SC
o2	Oxygen concentration
death signal	Mean and std of apoptotic and necrotic signals for NB and SC
Differentiation NB	Mean and std for the differentiation degree of NB
ratio_VEGF-NB_SC.txt	ratio of VEGF secreting NB vs SC
<b>Subcellular (CHMT)</b>	
<b>Input file</b>	<b>Description</b>
Ktrans	Vascularisation (particularised)
Effects of drug	Effects of drug, list of chemotherapy treatments
<b>Output files</b>	<b>Description</b>
drug_effects	Array with the inhibitions for each protein (inhibition probability value between 0 and 1).
start_effects	Array with the beginnings (hour format) of the effects of the drugs (> 1 position in case the cycles have a duration <2 weeks)
end_effects	Array with the endings of the drug effects.