

A Modelling Framework for Vascular Tumour Growth: An Extension to the Chaste Open Source C++ Library

Anthony J Connor^{1,2}, James A Grogan¹, Philip K Maini¹, Helen M Byrne^{1,2}, and Joe Pitt-Francis²

¹Wolfson Centre for Mathematical Biology, Mathematical Institute, University of Oxford, Oxford, OX2 6GG, UK.

²Department of Computer Science, University of Oxford, Oxford, OX1 3QD, UK.

Abstract

Chaste – Cancer, Heart And Soft Tissue Environment – is an open source C++ library for the computational simulation of mathematical models developed for physiology and biology. Code development has been driven by three applications: cardiac electrophysiology, cancer and developmental biology. The latter two applications have lead to the previous development of discrete cell modelling functionality, including cell cycling described by ordinary differential equations and chemical transport described by partial differential equations. An extension for simulating vascular tumour growth has been developed, including models of off- and on-lattice angiogenesis, blood flow and structural adaptation and solute transport in vessels. These new developments extend the ability of Chaste to multi-scale modelling of the growth and treatment of vascularized tissues, with particular focus on tumours. The software is available as an open source (Berkeley Software Distribution license) Chaste extension project, written in C++, but with Python bindings for rapid model building and exploration. Code functionality and design is over-viewed and example applications, including simulation of a well known 'snail-trail' problem and a three-dimensional, off-lattice corneal micro-pocket assay problem, are demonstrated.

Keywords: Chaste - agent-based simulation - multi-scale model - vascular tumour growth - on-lattice model - off-lattice model

AUTHOR SUMMARY:

- Insert a few bullets in here explaining major contributions of paper.

Contents

1	Introduction	3
2	Design and implementation	4
3	Results and exemplar simulations	4
3.1	Avascular tumour spheroid growth	4
3.1.1	On-lattice	4
3.1.2	Off-lattice	4
3.2	Vascular tumour growth	4
3.2.1	On-lattice	4
3.2.2	Off-lattice	4
3.3	An off-lattice model of corneal angiogenesis on a complex domain	4
3.4	Modularity	4
4	Discussion and future work	4

1 Introduction

There now exists a range of open-source frameworks for agent based and multi-scale modelling of soft tissues, including Chaste [1], OpenCMISS [2] and CompuCell3D [3]. These software do not yet have functionality for modelling tissue with growing vasculature, as is of interest when modelling many tumours. Although there are many mathematical models of vascular tumour growth [4], with an increasing number being released in the form of bespoke, open source software [5], a well documented and tested framework in which a range of models can be quickly built and predictions cross-compared does not yet exist.

The benefits of flexible and extensible software development in discrete soft tissue modelling are well known [6]. Modularity is of particular consequence for multi-scale models of vascular tumour growth. These models are usually comprised of several interacting sub-models, which include the effects of blood flow, cell cycling, nutrient transport and angiogenesis [7]. There are many ways to formulate the sub-models, incorporating different biophysical details [8] and many questions remain regarding the sensitivity of overall model predictions to the chosen combination of sub-models [9]. Indeed, it has been noted that cross-comparing the predictions of the many different models of vascular tumour growth is an important, and challenging, step for the field of tumour modelling as a whole [10].

The aim of the current project is to develop a modular framework for constructing and cross-comparing models of vascular tumour growth. A balance of performance, necessary when solving coupled problems over multiple size-scales, and the ability to readily construct models when exploring large model spaces, is achieved by using C++ code with Python bindings. Python binding of the code has three further, significant, advantages: i) users can quickly modify much of the functionality of a model in Python, without the need for recompilation, through Python-C++ inheritance and 'modifier' patterns, ii) external finite element solvers such as FEniCS and Abaqus can be readily incorporated, allowing challenging multi-physics and mechanics problems to be solved as sub-models of the tumour growth model, iii) use of different mark-up languages to describe models, for example SBML for the cell cycle, is straight-forward thanks to the availability of many Python tools [11]. The project is closely integrated with the Chaste C++ library for discrete cell modelling, sharing a new CMAKE based build-system, testing and documentation practices [12] and has a subset of the dependencies, namely VTK for input, output and geometry calculations, PETSc and Boost uBLAS for vector and matrix operations and CVODE for solving systems of ordinary differential equations.

Thanks to the aforementioned developments, and integration with Chaste, the models of vascular tumour growth included in this release, and presented as sample applications, represent the state-of-the-art in the field. Presented applications range from well known snail-trail models [13], allowing comparison with results of previous studies, through to three dimensional off-lattice angiogenesis models on complex geometries with mechanical interaction between discrete cells.

The structure of the report is now over-viewed. In Section 2 code design and implementation details are

introduced. The general design of Chaste is briefly summarized, however the reader should refer to [1] for further details. In Section 3 sample applications, including typical angiogenesis and vascular tumour growth problems are demonstrated, solved with a variety of solution techniques. While the focus is on cell centred based vascular tumour growth, it is noted that the framework is widely applicable to vascular tissues, and can be used with other functionality in Chaste, including vertex and Cellular Potts type simulations. In Section 4 code availability, future directions and other potential applications are discussed.

2 Design and implementation

- Describe how the cell based simulation classes are implemented so that they are extensible and easily customisable. In particular we should aim to describe how the template method pattern and strategy pattern are employed within the Solve method (the modifiers are essentially glorified strategies).
- Emphasise composability of simulations.
- Code chart.

3 Results and exemplar simulations

3.1 Avascular tumour spheroid growth

3.1.1 On-lattice

3.1.2 Off-lattice

3.2 Vascular tumour growth

3.2.1 On-lattice

3.2.2 Off-lattice

3.3 An off-lattice model of corneal angiogenesis on a complex domain

3.4 Modularity

Modularity and how it helps with validation and verification.

4 Discussion and future work