The influence of cell level model choice on the outcome of multiscale simulations

J.M. Osborne[1,2,†*], A.G. Fletcher[2,3,†], P.K. Maini[2,3] and D.J. Gavaghan[1,2] + TBC

[1] *Oxford University Computing Laboratory, Parks Road, Oxford OX1 3QD, UK*
[2] *Oxford University Department of Biochemistry, South Parks Road, Oxford OX1 3QU, UK*
[3] *Oxford University Mathematical Institute, 24-29 St Giles', Oxford OX1 3LB, UK*
[†] *Joint first authors.*

Abstract

[Ozzy: The key results intended for this paper are:

- **A comparison of different cell-level modelling approaches (cellular Potts model, centre dynamics models and vertex dynamics models)**
- **Brief discussion of the implementation of these approaches within the Chaste framework**
- **Comparison of the different Chaste codes used to create example simulations**

Possible journal is SIAM: Multiscale Modeling and Simulation]

[*]Corresponding author (James.Osborne@comlab.ox.ac.uk)

Contents

1 Introduction [Still to do]

- Short review of cell-based models
- Where models come from
- simulation methods
- Why we need a consistent simulation framework

[**Alex: What other comparison papers (if any) have been written?**]
The remainder of this paper is structured as follows...

2 Cell based models of biological tissues [Still to do]

- CA
- Potts
- Cell centre
- Cell vertex

3 Framework structure [Started]

The multiscale modelling framework consists of three main interacting scales: the cellular scale; the subcellular scale; and the tissue scale.

The cellular (meso) scale. The cell is the central element of the framework.

Cellular Automata. The simplest form of cellular modelling, cells are represented as points on a lattice and rules constructed in order for each cell to divide, move or die. Cells can only divide in to neighbouring spaces on the lattice. Some of the issues with this type of modelling include: effects felt at distance; fixed regular lattice geometry; growth is lattice dependent – 4 point $\rightarrow$ star and 8 point $\rightarrow$ octagon; and finally you can't consider stress in the tissue.

Cellular Potts. An extension of cellular automata model is the Cellular Potts Model, in which cells are composed of a number of lattice points, and the cell shape is evolved by minimizing the energy which is calculated by imposing constraints due to chemotaxis etc. This form of modelling is utilised in the CompuCell3D modelling package produced at Indiana University.

Overlapping sphere models. See the work of Drasdo *et al*, for example [1]. Cells are modelled as overlapping spheres and energy minimization is used to evolve the system, cells can divide depending on binary rules or buy using a cell cycle model. The software package MASON from UCSF deals with this type of modelling. Extensions include ellipses with varying radii and orientation. Chaste functionality for this is discussed in a recent computational study by Pathmanathan et al. [5] of the mechanical properties of this class of model.

Subcellular Element Method. This is an extension of the overlapping spheres approach and treats cells as being composed of subcellular elements with different interactions between inter-cell and intra-cell elements [3, 8]

Centre Dynamics Tessellation Models. This was the first class of cell-based model added to Chaste [10] and are based on the model presented by Meineke et al. [2].

Vertex Dynamics Models. Cells defined by polygons and movement determined by energy minimization. Currently implemented in Chaste in two dimensions [4].

Finite Element Models. Generalization of vertex where cell represented as collections of moving and deforming elements – tetrahedra for example.

The subcellular (micro) scale. The behaviour of cells is dictated by modelling the numerous subcellular processes occurring within the cell. The main modelling paradigm is to use reaction networks rendered into ODEs which interacting with cell level through the cell cycle and also directly through properties such as cell size, shape and motility etc. Other methods include: stochastic ODEs; PDEs; or fully stochastic networks which representing the transport and use of substrates within cells.

The tissue (macro) scale. Modelling of inter-cellular signalling, or nutrient transfer can be considered by using reaction diffusion equations, which are solved on the growing domain (defined by the cells) with cells acting as sinks and sources [**Ozzy: We don't have PDEs for vertex yet: we should reference Aaron's paper here** [**9**]].

The advantage of such a framework is that once you have implemented a particular subcellular model you can use it in all possible cell level models.

3.1 Computational structure [Started]

Chaste (Cancer, Heart and Soft Tissue Environment) is a collaborative software development project, which is designed to act as a high-quality multi-purpose library supporting computational simulations for a wide range of biological problems [6]. In this context, high-quality means that the software is extensible, robust, fast, accurate and maintainable and uses state-of-the-art numerical techniques. It is also open-source, so can be adapted by other developers. Chaste has been developed by a multidisciplinary team including mathematicians and software engineers. This ensures that the code is well-structured as a piece of software, while at the same time practical and useful as a computational modelling tool. While it is a generic extensible library, to date attention has focused on the fields of cardiac electrophysiology and tumor growth [7]. Further details on Chaste are available at `http://web.comlab.ox.ac.uk/chaste`.

3.2 Structure of code [Started]

To set up a cell-based simulation within Chaste you need to instantiate a collection of objects which specify the properties of the simulation. The following seven objects can

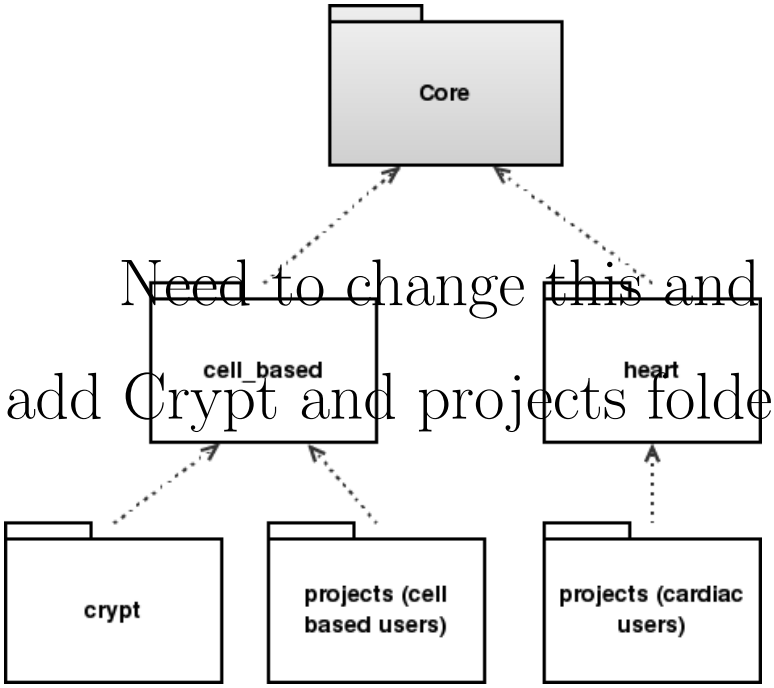

Figure 1: Schematic diagram of Chaste code base.

be defined for a simulation to be undertaken:

Mesh, the structural information for the simulation, the initial epithelial layer layout is defined here.

Cells, a collection of cells to be associated with the mesh. Contain all the subcellular machinery. The initial conditions for subcellular models are defined here.

Cell population, associated the cells with the mesh and provides a mechanism for cells to know their location.

Simulation, takes in the cell population and implements the rules for evolution of the system.

Forces, these are passed in to the simulation and dictate the way the vertices move. This is the mechanism for selection of the type of vertex dynamics model. Multiple forces can be applied allowing the introduction of processes like chemotaxis.

***Cell killers,** these are passed to the simulation and define how cells die. Multiple killers can be used at the same time for example to represent random death and death due to sloughing.

***Boundary Conditions (BCS),** these are passed to the simulation and define any fixed restrictions on how the cells can move. Multiple conditions can be used at the same time.

The stared objects are optional as you may not want to have cells being removed from the simulation and the growth may be unconstrained.

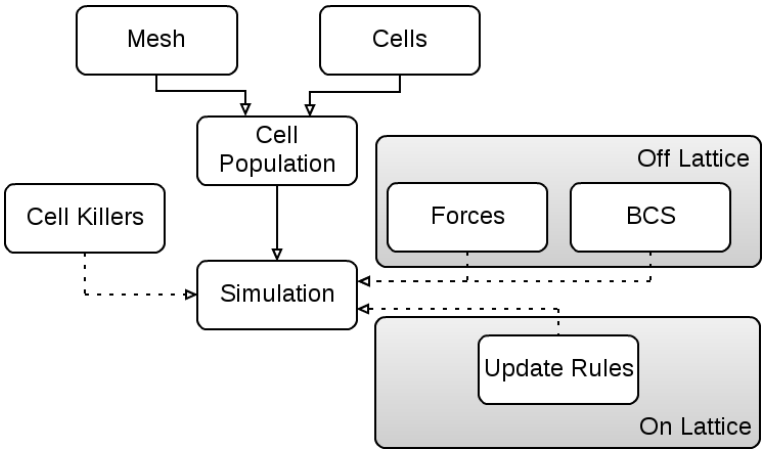

Figure 2: Schematic diagram of the framework showing the separate components of the simulations and the communication between the classes.

4 Example simulations [Started]

- Monolayer
- Spheroid
- crypt??

4.1 Demonstration of growing monolayer [Started]

- Demonstrate basic functionality
- Example of growing monolayer with void removal, sloughing? etc.
- Show differneces in code from model to model

```
1  void TestVertexMonolayerWithRandomBirthAndDeath() throw (Exception)
2  {
3      // Create a simple 2D vertex mesh
4      HoneycombVertexMeshGenerator generator(5, 5);
5      MutableVertexMesh<2,2>* p_mesh = generator.GetMesh();
6
7      // Set up cells, one for each element in the vertex mesh
8      std:vector<CellPtr> cells;
9      CellsGenerator<StochasticDurationGenerationBasedCellCycleModel, 2>
10         cells_generator;
11     cells_generator.GenerateBasic(cells,p_mesh->GetNumElements());
12
13     // Create cell population
```

```
    VertexBasedCellPopulation<2> cell_population(*p_mesh, cells);

    // Set up cell-based simulation
    CellBasedSimulation<2> simulator(cell_population);
    simulator.SetOutputDirectory("TestVertexMonolayer");
    simulator.SetEndTime(40);

    // Create a force law and pass it to the simulation
    NagaiHondaForce<2> force_law;
    simulator.AddForce(&force_law);

    // Create cell killer and pass it to the simulation
    RandomCellKiller<2> random_cell_killer(&cell_population, 0.01);
    simulator.AddCellKiller(&random_cell_killer);

    // Run simulation
    simulator.Solve();
}
```

4.2 Demonstration of cell sorting [Still to do]

- Potts and Vertex comparison.

4.3 Demonstration of spheroid [Still to do]

- Include nutrient diffusion.

4.4 Demonstration of crypt [Still to do]

- 2D cylindrical crypt in each modality.
- 3D node based crypt.

5 Discussion [Still to do]

Acknowledgements [Started]

JMO and AGF are funded through the OCISB project (BB/D020190/1). JMO is also supported by the Life Sciences Interface and Systems Biology Doctoral Training Centres (EP/E501605/1 and EP/G50029/1 respectively).

References

[1] Drasdo, D., Hoehme, S., and Block, M. (2007). On the role of physics in the growth and pattern formation of multi-cellular systems: What can we learn from individual-cell based models? *J. Stat. Phys.*, 128:287–345.

[2] Meineke, F. A., Potten, C. S., and Loeffler, M. (2001). Cell migration and organization in the intestinal crypt using a lattice-free model. *Cell Prolif.*, 34:253–266.

[3] Newman, T. J. (2005). Modeling multi-cellular systems using sub-cellular elements. *Math. Biosci. Eng.*, 2:611–622.

[4] Osborne, J., Walter, A., Kershaw, S., Mirams, G., Fletcher, A., Pathmanathan, P., Gavaghan, D., Jensen, O., Maini, P., and Byrne, H. (2010). A hybrid approach to multi-scale modelling of cancer. *Phil. Trans. Roy. Soc.A*, 368:5013–5028.

[5] Pathmanathan, P., Cooper, J., Fletcher, A., Mirams, G., Murray, P., Osborne, J., Pitt-Francis, J., Walter, A., and Chapman, S. (2009). A computational study of discrete mechanical tissue models. *Phys. Biol.*, 6:036001.

[6] Pitt-Francis, J., Bernabeu, M., Cooper, J., Garny, A., Momtahan, L., Osborne, J., Pathmanathan, P., Rodriguez, B., Whiteley, J., and Gavaghan, D. (2008). Chaste: using agile programming techniques to develop computational biology software. *Phil. Trans. Roy. Soc. A*, 366:3111–3136.

[7] Pitt-Francis, J., Pathmanathan, P., Bernabeu, M., Bordas, R., Cooper, J., Fletcher, A., Mirams, G., Murray, P., Osborne, J., Walter, A., Chapman, S., Garny, A., van Leeuwen, I., Maini, P., Rodriguez, B., Waters, S., Whiteley, J., Byrne, H., and Gavaghan, D. (2009). Chaste: a test-driven approach to software development for biological modelling. *Comp. Phys. Comm.*, 180(12):2452–2471.

[8] Sandersius, S. and Newman, T. (2008). Modeling cell rheology with the Subcellular Element Model. *Phys. Biol.*, 5:015002.

[9] Smith, A., Baker, R., Kay, D., and Maini, P. (2011). Incorporating chemical signalling factors into cell-based models of growing epithelial tissues. *J. Math. Biol.*, page Submitted.

[10] Van Leeuwen, I., Mirams, G., Walter, A., Fletcher, A., Murray, P., Osborne, J., Varma, S., Young, S., Cooper, J., Doyle, B., Pitt-Francis, J., Momtahan, L., Pathmanathan, P., Whiteley, J., Chapman, S., Gavaghan, D., Jensen, O., King, J., Maini, P., Waters, S., and Byrne, H. (2009). An integrative computational model for intestinal tissue renewal. *Cell Prolif.*, 42:617–636.