Hierarchical Reconstructions of Cardiac Tissue

A V Holden[1], M J Poole[2] and J V Tucker[2]

[1]Department of Physiology and Centre for Nonlinear Studies,
University of Leeds, Leeds LS2 9JT, England

[2]Department of Computer Science,
University of Wales Swansea, Swansea SA2 8PP, Wales

Abstract

We consider the general problem of comparing and integrating computational models of cardiac tissue at different levels of physiological detail. We use a general theory of synchronous concurrent algorithms to model spatially extended biological systems, and expand the theory to create hierarchical models by relating observable behaviour at different levels. The general concepts and methods are illustrated by a detailed study of electrical behaviour in cardiac tissue, in which models based on coupled systems of ordinary differential equations, partial differential equations and cellular automata are compared and combined.

1 Introduction

Biological systems are characterised both by their hierarchical organisation and their complexity. The physiology of an organ is determined by its tissue behaviour, generated by cellular and subcellular processes, but it may only emerge when the organ is integrated into a system within the organism. The structures and behaviours at any one of these levels are intrinsically complicated, and operate with different time and space scales. Basic questions arise: How are behaviours at different levels related and integrated?

A *qualitative* approach is no longer sufficient for integrative physiology; a *quantitative* approach supported by mathematical models and experimental data must be developed. Furthermore, these mathematical models are nonlinear and the sensitivity of nonlinear systems to changes in parameter values (where close to a bifurcation point small changes can produce qualitative changes in behaviour) require quantitative approaches for their proper understanding. The complexity and nonlinearity forces such a quantitative approach to be based on computational mathematical models (i.e. algorithms), rather than analytic mathematical models; see [10].

The minimal aim of any computational biology is the simulation of biological processes. The aim of an *integrative* computational biology is to provide understanding of how different behaviours emerge at different levels; how activity at one level in the hierarchy influences, and is influenced by, activity at other levels; and to tie together the large amount of detailed information about behaviours on different space and time scales into a cohesive whole. The biological problem to be solved by an integrative computational biology is the reconstruction of the system from the behaviours and interrelations of its components.

A computational approach to biology cannot succeed by brute force simulation, where the details of the biological components and their interactions are simulated all at the cellular or subcellular level. A cellular reconstruction of whole-organ behaviour, for example, is bound to fail, firstly because the amount of information involved would be overwhelming, and secondly due to the prohibitive computing costs, even allowing for foreseeable advances in computing technology.

The hierarchical nature of biological systems, with behaviour at different levels characterised by different variables, needs to be retained. As an example, the intracellular activity of a heart cell is described by spatially distributed concentrations of various ionic species; the cellular activity by membrane potential and contractive state; the behaviour of cardiac tissue by travelling waves of excitation and stress and strain tensors; the activity of ventricles by their behaviour as a pump, with cardiac output and systemic arterial pressure as the physiologically significant output. To reproduce the beating of the heart, all that is required is a description of when different regions of the heart are excited. Thus, for whole-heart dynamics, a simplified and abstract representation of excitation can be sufficient. Reconstruction of intracellular and cellular behaviours are obviously necessary, but are meaningful only for small regions of tissue within the whole-heart. Thus we need a computational approach to modelling that uses different, interacting models to represent a system with different, interacting layers.

Different types of mathematical structures are used to model the processes and behaviours at

different levels. Some will be spatially extended systems, some time series; some will be discrete, some continuous. In the modelling of spatially extended systems, for example, partial differential equations (PDEs), coupled ordinary differential equations (CODEs) and cellular automata (CAs) are commonly used. When implemented or approximated as algorithms on a computer, all of the models have discrete space and time, where the spatial- and temporal-granularity is related to the level of the reconstructed behaviour (see Figure 1).

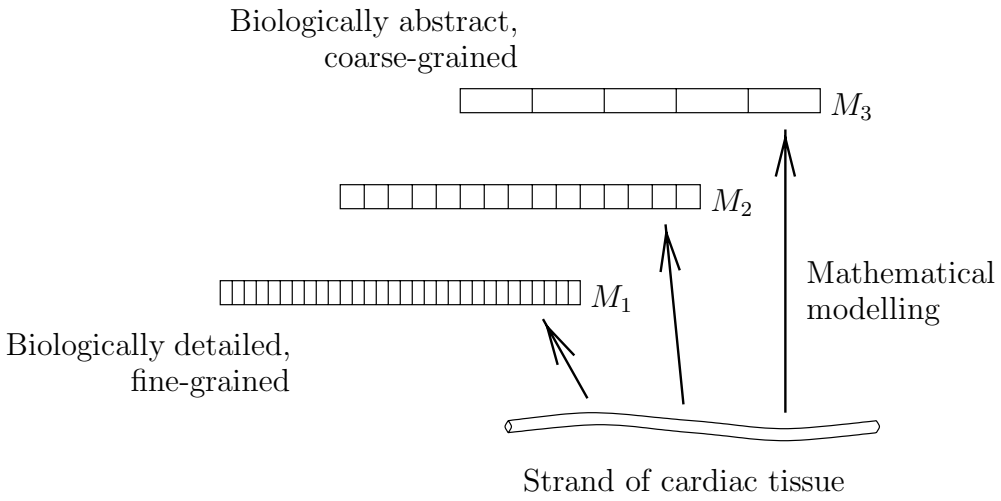

Figure 1: A strand of cardiac tissue reconstructed at three levels by algorithmic models M_1, M_2, M_3, showing schematically the granularities of the models' discrete spaces.

A model of an overall integrative system will be a system of interacting models of different mathematical type—a multi-level mathematically hybrid form of model. In order to construct meaningful multi-level models, concepts of abstraction and approximation between single-level models needs to be clarified and validated. Thus we require a framework within which different component models can be specified, compared and integrated. In [17],[13],[15] and [14] we have argued that the *theory of synchronous concurrent algorithms* (SCAs) provides a common framework for specifying models of excitable tissues. Here we will extend SCA theory to accommodate concepts of abstraction and approximation for models of cardiac tissue, and for building integrated multi-level models. A formal concept of approximation of behaviours between two models will be defined, built upon abstractions between the models' discrete spaces and times, and, particularly, on approximations of local states of elements of the models related by space abstraction. This allows the classification of models where the behaviour of each model approximates the behaviour of the model beneath it (see Figure 3). It is the abstractions of components and the approximations of behaviours that allows the construction of *hierarchical* or *multi-level* models by substitution of spatially-related elements of models (see Figure 3).

Here we use the theory of SCAs to construct and analyse models of cardiac tissue; this is selected as a case study since we can clearly define the behaviour we are interested in—the propagation of electrical activity—and because of the simplicity of the cell-to-cell interactions, by a diffusive nearest neighbour interaction. We compare three different types of model, and construct hybrid

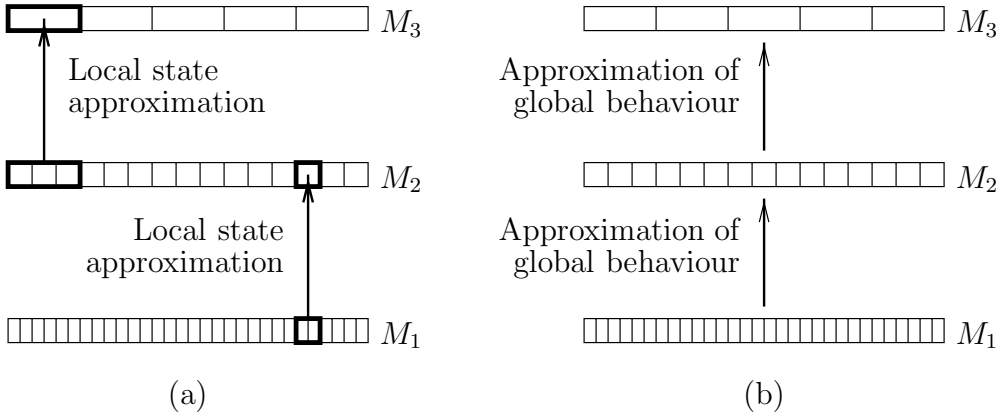

Figure 2: Models M_1, M_2, M_3 classified as a hierarchy. (a) The comparisons between the models are defined in terms of comparisons of local states. (b) M_2 approximates M_1 and M_3 approximates M_2.

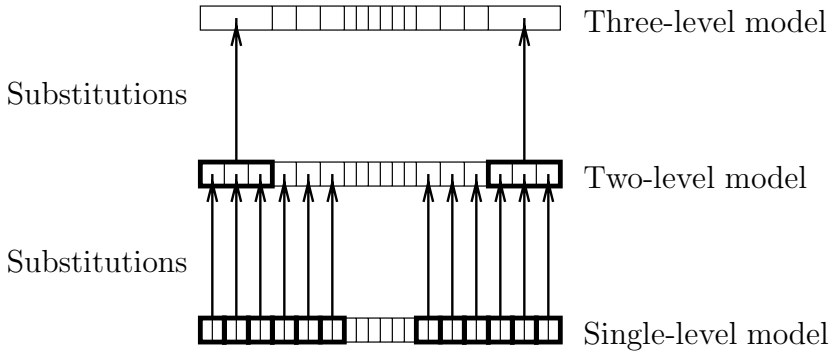

Figure 3: Construction of a multi-level model beginning with the most biologically detailed, fine-grained model M_1, through a process of substituting models' elements with their abstractions, we achieve a three-level model. The central region of a strand of cardiac tissue is modelled using elements from M_1; this is surrounded on either side by regions modelled by M_2; and the boundary regions are modelled at the most abstract level, by elements of M_3. Communication between elements from different models is determined by the time, space and local state abstractions used, in Figure 2 to compare the models.

multilevel models from these components. This case study is used to develop and illustrate concepts of abstraction within hierarchical or multi-level models. The three models are:

(i) a lattice of coupled biophysically derived ODE models of single cardiac cells;

(ii) a PDE model of electrical wave propagation that approximates some behaviours specific to cardiac excitation; and

(iii) a CA model that gives basic excitability and wave-propagation properties.

The structure of the paper is as follows. In Section 2 we discuss the modelling of electrical

activity in the heart, emphasising hierarchical aspects defined by functionality and mathematical models. In Section 3 we describe the general framework of synchronous concurrent algorithms in the specific context of modelling the heart using different types of mathematical model. In Section 4 we introduce CODE, PDE and CA models of cardiac tissue that we will use to illustrate our theory, each describing electrical behaviour at a different level of biological detail and operating on different time- and space scales. Each model is transformed into an SCA and its basic behaviours illustrated and compared with those of the other models. In Section 5 we introduce formal notions of abstraction between components of SCA models of cardiac tissue, and approximation between their behaviours. In Section 6 we illustrate these concepts of abstraction and approximation using the three models defined in Section 4. In Section 7 we define a general method for building hybrid multi-level models from a hierarchy of single-level models. In Section 8 we illustrate this construction technique models by building, and analysing the behaviour of, two- and three-level models build from the component models of Section 4 and the comparison in Section 6. Finally, in Section 8, we provide some concluding remarks and directions for further work.

2 Hierarchical models of the heart

The mechanical pumping action of the heart drives the pulsatile and sometimes turbulent pumping of blood, and is triggered by waves of excitation propagating through the cardiac muscle. Thus, the heart is a structured, 3-dimensional object that moves as it contracts, and whose motion is coupled to fluid flows. Here we will just consider the propagation of waves of electrical activity through the heart, and ignore the moving geometry and the ways in which it can alter the pattern of propagation. Propagation of activity through cardiac tissue is by local circuit current flow through the cell-to-cell coupling, and this can be represented by linear coupling between neighbouring cells, or, in a continuous representation, by a diffusive interaction. However, the heart is not an isotropic and homogeneous medium. The propagation velocity is up to three times faster along the axis of ventricular cells than transverse to the axis, and so a representation of the structure of the electrical geometry of the heart is 3-dimensional tensor of coupling or diffusion coefficients. The pattern of propagation in an anatomically realistic heart model is complicated, by both the geometry of the heart and the regional difference in excitation properties [26, 12].

2.1 Hierarchy of models

Propagating waves of electrical activity in the heart may be modelled in a number of ways:

(i) by spatially and temporally continuous systems, such as partial differential equations. These PDEs may be simple caricatures (e.g., the *FitzHugh-Nagumo* equations [7]) or complicated systems that provide a biophysically detailed description of the wave and its controlling processes (e.g., for ventricular tissue [4]);

(ii) by spatially discrete and temporally continuous systems, such as lattices of coupled (simple, or biophysically accurate) ordinary differential equations (CODE lattices);

(iii) by spatially and temporally discrete systems, such as coupled map lattice (CML) of cellular automata (CA) models; and

(iv) by the local properties of excitability defined by ordinary differential equations, such as those in the Oxsoft suite [23].

If $u(x,t)$ is the observable state at site x at time t (membrane potential in the context of cardiac tissue), these different types of model describe wave phenomena at different levels of abstraction from the physical behaviour: the PDEs of (i), CODEs of (ii) and CMLs of (iii) model the spatio-temporal patterning of $u(x,t)$; and the ODEs of (iv) model time series at a point x in the shape of action potentials due to their change in rate or interactions between different travelling waves.

In computational investigations, the type of model used is based not only on the problem being investigated, but also on the available computing power: the Oxsoft ODE models [23] can run on a personal computer, simple PDE models [25] can run on a workstation, whereas biophysically accurate PDE models require higher performance machines and supercomputers [6, 2, 3]; earlier work on CODE lattice models required a Connection Machine [34].

From the mathematical viewpoint, the PDE models are "best" in the sense that their theory is best developed [11]. From the biological viewpoint the CODE model is "best" in the sense that it preserves the cellularity of cardiac tissue. Such CODE lattices do exhibit behaviours seen in experiments and not seen in continuum models [19]. However, rather than considering one type of model as correct, and the rest as approximations, we consider the models as different but inter-related and forming a hierarchy of models, where the different levels provide computationally efficient descriptions of different behaviours. For example, consider the interaction between two colliding waves by solutions of PDEs, and rate dependent changes in action potential shape by the solution of ODE models. These levels are inter-related. A CA model can provide the location of wavefronts for a phenomenological PDE model. The PDE model, in turn, provides an approximate global spatial pattern of excitation for a biophysically accurate local PDE or CODE lattice model, or ODE model of activity at a point. Conversely, changes in action potential shape generated by the local model will alter parameters that control action potential duration in the CA or phenomenological PDE model.

We propose to model this dynamical functional cardiac hierarchy by a computational hierarchy, which models the following:

(i) cell behaviour by a system of high order ODEs (e.g., components of the Oxsoft suite [23]);

(ii) local tissue behaviour (corresponding to a cuboid of less than one cubic centimetre volume of myocardium) by biophysically derived CODE lattices and PDEs;

(iii) larger scale local behaviour by PDEs (derived from the biophysical ODEs by separating fast and slow processes), by phenomenological systems of PDEs (e.g., the FitzHugh-Nagumo equations); and

(iv) whole-heart behaviour by sculptured composite CA and phenomenological systems of PDEs in which anisotropy is represented by a diffusion or coupling tensor.

The common mathematical framework of the theory of SCAs needed to study this hierarchy of models types will be defined and illustrated in Section 3, and applied to CODE, PDE and CA models in Section 4 and to multi-level models in Section 8.

2.2 Mathematical models of excitable media

Cardiac tissue is a structured, anisotropic functional system of electrically coupled cells. Its electrical behaviour may be idealised by considering activity in a bounded two- or three-dimensional excitable medium, whose local properties represent the nonlinear behaviour of an excitable cell, and one of whose kinetic variables (membrane potential) interacts diffusively. Such an excitable medium responds to a small, localised perturbation by a local, decremental response, but if the perturbation is sufficiently large (in intensity, duration and spatial extent), i.e. above a threshold, the response is a large amplitude travelling wave. Such an idealised excitable medium can be modelled by a nonlinear, parabolic partial differential equation

$$\frac{\partial \mathbf{u}(x,t)}{\partial t} = \mathbf{K}(\mathbf{u}(x,t)) + D\nabla^2\mathbf{u}(x,t)$$

where $\mathbf{u}(x,t) \in \mathbf{R}^m$, the m-dimensional state space, $x \in \mathbf{R}^2$ or $\mathbf{R}^3$, 2- or 3-dimensional physical space, and D is a matrix of real valued, non-negative diffusion coefficients. If fibre orientation (by means of methods of [22]) is used to give anisotropy in conduction velocity then D is a tensor. The function $\mathbf{K}(\mathbf{u})$ may involve simple polynomial terms, as in the FitzHugh-Nagumo equations, or may be quite cumbersome, as in biophysically derived excitation equations. For cardiac tissue only one of the diffusion coefficients is nonzero [26].

Although cardiac tissue is a functional syncytium (i.e. electrical activity can be represented on a continuum) some propagation phenomena result from the coupling of discrete elements; namely cells or aggregates of cells [19]. This granularity of cardiac tissue may be modelled by a regular spatial lattice, in which the behaviour of each element is described by an ordinary differential equation

$$\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \mathbf{K}(\mathbf{u}(x,t))$$

to form a CODE lattice where, as in the PDE model above, the function $\mathbf{K}$ represents the membrane ionic currents [34]. The behaviour of a single cell may be represented by a single system of ODEs, as in the Oxsoft package [23].

If time and space is discretised, $t \in \mathbf{N}$, $x \in \mathbf{Z}^2$ or $\mathbf{Z}^3$ ranges over a two- or three-dimensional integer lattice, a CA model can be used. CAs provide a computationally more efficient, but theoretically less well justified, alternative to PDE models. In terms of cardiac cellular electrophysiology, a CA model is too radical a caricature of cellular excitability to be taken seriously. However, a CA model coupled with an appropriate anisotropic geometry provides a means for following the propagation of wavefronts.

Thus there are a number of different, interacting levels, each of which may be a different kind of mathematical structure: ordinary or partial differential equations and cellular automata, or a variable derived from the evolution of one of these models, such as the beat-beat interval. These different models have different structures in finite or infinite dimensional state spaces, and have different timescales. Reconstructing the electrical activity of the heart, by both simulating its spatio-temporal patterning and abstracting its features, requires coupling between these models. To couple different types of model we require a common theoretical structure, as is provided by the theory of synchronous concurrent algorithms.

3 A framework for algorithmic models of the heart

Models of cardiac activity range from descriptions of propagation in cardiac tissue and excitation-contraction coupling, to the mechanical activity of the heart [26]. These different types of model are organised in a functional and causal hierarchy, in that cellular excitation leads to propagation, and is triggered by propagation; excitation leads to contraction via excitation -contraction coupling, and is modified by contraction via mechanoelectric feedback; and tissue contraction leads to the beating of the heart and its functioning as a pump. Thus models of different processes, that act on different time and space scales, need to be coupled. The construction of a computational model of the heart is by coupling different types of component models, within a hierarchy, that progressively moves from detailed, local dynamics to large scale global behaviour. To approach different models—ODES and PDES, for biophysically realistic [23, 2, 3, 4], or simplified models with appropriate geometry [14], we need a framework that can couple different types of model.

These mathematical models can be unified and integrated through their representations as algorithms. It turns out that all the models become algorithms based on discrete space and time (though with discrete or continuous state) with a common structure: they are all synchronous concurrent algorithms.

3.1 Synchronous concurrent algorithms

A *synchronous concurrent algorithm* (SCA) is a spatially distributed network of computing units called *modules*, and connections called *channels*, that compute and communicate in parallel on data from an arbitrary set A and are synchronised by a global clock $T = \{0, 1, 2, \ldots\}$ measuring discrete time. A network computes given an initial state and infinite sequences or *streams* of data that are input or parameter changes. The concept of an SCA has been introduced in theoretical computer science [9, 29] to model parallel deterministic computing systems, especially hardware. The parallelism and determinism of these algorithms are explicitly formulated in terms of discrete space and discrete time. Many mathematical models of excitable media and cardiac tissue have been shown to be SCAs [17], including cellular automata, coupled map lattices, and discrete approximations of partial differential equations and coupled ordinary differential equations.

SCA theory is built upon the theory of primitive recursive functions over many-sorted algebras

[30, 31, 32]. Thus SCAs provide us with a mathematically rigorous framework for the investigation of disparate models of cardiac tissue in a unified way. Specifically, SCAs are suited to (i) the comparison, composition and decomposition of models, including the construction of hybrid and multilevel models [13, 16, 27]; (ii) experimental work with simulation software [20, 5]; and (iii) the classification of the computational behaviour of models of parallel deterministic systems, including the limits of computation [21, 18].

In the present paper, we use the theory of SCAs as a framework and toolset for comparing, composing different types of cellular and tissue models in a hierarchical, structured organ model, and so as a general approach towards a computational integrative biology.

3.2 SCA models of cardiac tissue

We describe formally the general structure of SCA cardiac tissue models. An SCA is characterised by the physical *space* over which it computes, its *timing* properties, the *data* used to describe tissue state and input parameters, and the *local dynamics* of each point or cell within the space. In Section 3.3 we shall consider the specific case of one-dimensional SCA models which we will use later to illustrate our general methods.

Space. Let I be a finite set of points or *computational "cells"* in one- two- or three-dimensions, where each cell $i \in I$ represents a region of cardiac tissue. Computational cells may represent individual cardiac cells (in the case of fine-grained, biologically detailed models) or lengths, areas or volumes of tissue composed from a number of cardiac cells in less-detailed models, where the number of cells represented is related to the spatial granularity of the model.

For each cell $i \in I$ let

$$nhd(i) = \{r(i,1), r(i,2), \ldots, r(i,p(i))\} \subseteq I$$

be a set of *local neighbours* of i (enumerated by *architecture functions* $p : I \to \mathbf{N}$ and $r : I \times \mathbf{N} \to I$). States of the cells in the neighbourhood $nhd(i)$ are inputs to the computation of the state of i at each timestep of the algorithm. Figure 4 illustrates a cell on the surface of the heart surrounded by a typical local neighbourhood.

Time. Let $T = \{0, 1, 2, \ldots\}$ be a discrete clock over which the algorithm computes. For fine-grained, biologically detailed models, each clock cycle is likely to represent a period of time of the order of 10's of microseconds. For more abstract models, the clock is slower: a clock cycle will typically represent a period of time in the order of milliseconds.

Tissue state. Let the state of each computational cell at any timestep comprise two components:

(i) an *observable* part that represents the *excitation* or *voltage* of the cardiac tissue, and is communicated at each timestep to all cells for which the cell is a neighbour; and

(ii) a *hidden* part that remains local to the cell. This state represents different properties for different models: for detailed, biophysically derived models it may include concentrations

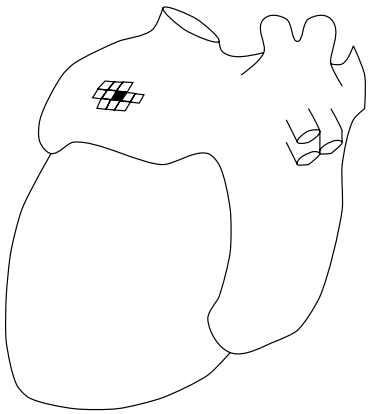

Figure 4: A computational cell on the surface of the heart (shown in black) representing a small volume of atrial myocardium, surrounded by a local neighbourhood of cells (shown in white).

of various ionic species, gating variables etc; for more abstract models it commonly represents a notion of "tissue recovery".

Formally, let

$$S = A \times B$$

be the set of data used to describe the states of cells where A is the set of all possible observable states and B is the set of all possible hidden states. The state of any cell is thus described by a pair of the form

(v, u) where $v \in A$ is its observable state and $u \in B$ its hidden state.

We shall assume that the set S is used to describe the possible states of every cell uniformly throughout the model. The global state of the model can also be represented by a pair of the form $(v, u) \in A^I \times B^I$ where for each cell $i \in I$, $v(i)$ and $u(i)$ are the observable and hidden states of i respectively.

Parameter streams. Let P be a set of all possible parameter values or external inputs to cells. Parameter sets commonly include data representing electrical stimulation; they may also include, for detailed models, effects of pharmacological agents and concentration changes produced by, for example, ischaemia or acidosis. Each cell $i \in I$ is supplied with a stream of the form $a_i : T \to A$ of parameter values from the set $[T \to P]$ of all streams of data from P clocked by T. The SCA is thus supplied with a set of streams of the form $a \in [T \to P]^I$ where $a_i(t)$ is the parameter value supplied to cell $i \in I$ at time $t \in T$.

Local dynamics. Cell behaviours are determined by their local dynamics. At each timestep $t \in T$, each cell $i \in I$ computes its next state (i.e. its state at time $t + 1$) in terms of its current state, the observable states of its $p(i)$ neighbours from the set $nhd(i)$, and the current parameter value $a_i(t)$ supplied to it. Formally, cell i computes a function

$$f_i : S \times A^{p(i)} \times P \to S$$

where $f_i(v, u, v_1, v_2, \ldots, v_{p(i)}, a)$ is the next state of cell i given its current state $(v, u) \in S$, its neighbours' current observable states (voltages) $v_1, v_2, \ldots, v_{p(i)} \in A$, and its current parameter value $a \in P$.

SCA equations. An SCA is completely specified from the above components, which we conveniently collect together into a tuple

$$M = (I, p, r, T, A, B, P, \langle f_i \mid i \in I \rangle).$$

comprising space I, architecture functions p and r, clock T, state sets A and B, parameter set P and family $\langle f_i \mid i \in I \rangle$ of local dynamics. From the components, we can determine the state of each cell $i \in I$ at any time $t \in T$, given an initial global state $(v, u) \in A^I \times B^I$ and a set $a \in [T \to P]^I$ of parameter streams. We accomplish this by defining *local state functions*

$$F_i : T \times [T \to P]^I \times A^I \times B^I \to S$$

for each cell $i \in I$, with the intention that

> $F_i(t, a, v, u) \in S$ is the state of cell i at time t given input parameter stream set a and global initial state (v, u).

The state $F_i(t, a, v, u) \in S$ comprises an observable part and a hidden part. We will find it useful express the former as a function

$$V_i : T \times [T \to P]^I \times A^I \times B^I \to A$$

such that for all t, a and (v, u), $F_i(t, a, v, u) = (V_i(t, a, v, u), u')$ for some hidden state $u' \in B^I$.

We define each F_i, and hence V_i, for all $a \in [T \to P]^I$, $v \in A^I$, $u \in B^I$, by induction on the clock T as follows:

$$\begin{aligned} F_i(0, a, v, u) &= (v(i), u(i)) \\ F_i(t+1, a, v, u) &= f_i(F_i(t, a, v, u), V_{r(i,1)}(t, a, v, u), \ldots, V_{r(i,p(i))}(t, a, v, u), a_i(t)). \end{aligned}$$

3.3 One-dimensional SCA models of cardiac tissue

Whilst the theory and techniques presented in this paper will be general in terms of model geometry, all the examples and case-studies given will, for clarity of definition and illustration, be one-dimensional with nearest neighbour architectures. Such one-dimensional models retain the features of planar wave propagation—propagation velocity of a solitary wavefront, rate dependence and parameter dependence of propagation velocity, and annihilation on collision. They also caricature the normal propagation of part of a wavefront in the heart from the sino-atrial node, through the atria, atrio-ventricular node, Purkinje fibres and ventricular wall.

Formally, we assume that space is indexed by a set $I = \{1, \ldots, n\}$ for some $n \geq 2$, and that the neighbourhood sets are $nhd(1) = \{2\}$, $nhd(n) = \{n-1\}$ and $nhd(i) = \{i-1, i+1\}$ for $i = 2, \ldots, n-1$

which are easily enumerated by architecture functions $p : I \to \mathbf{N}$ and $r : I \times \mathbf{N} \to I$ defined such that

$$\begin{array}{rclcrclcrcl} p(1) & = & 1 & & r(1,1) & = & 2 & & & & \\ p(n) & = & 1 & & r(n,1) & = & n-1 & & & & \\ p(i) & = & 2 & & r(i,1) & = & i-1 & & r(i,2) & = & i+1 \end{array} \qquad \text{for } 2 \leq i \leq n-1.$$

We illustrate the structure of such a model in Figure 5.

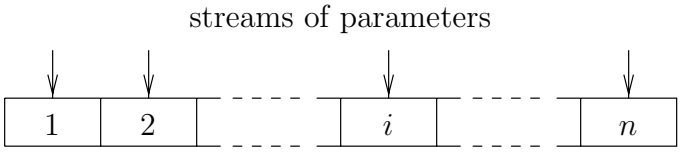

Figure 5: The structure of the one-dimensional cardiac tissue models presented in the paper.

Each non-end cell $i = 2, \ldots, n$ computes local dynamics of the form $f_i : S \times A^2 \times P \to S$ where $f_i(v, u, v_l, v_r, a)$ is the next state of i given current state $(v, u) \in S$, left and right neighbours' voltages $v_l, v_r \in A$ and parameter value $a \in P$. The end cells $i = 1, n$ compute local dynamics of the form $f_i : S \times A \times P \to S$ where $f_i(v, u, v_{l/r}, a)$ is the next-state of the cell given current state (v, u), left *or* right neighbouring voltage $v_{l/r}$ and parameter value a.

The local state functions

$$F_i : T \times [T \to P]^I \times A^I \times B^I \to S$$

for each cell $i \in i$ in the one-dimensional model are defined, for all sets $a \in [T \to P]^I$ of input streams and initial global states $(v, u) \in A^I \times B^I$, as follows:

At time 0. For each cell $i \in I$,

$$F_i(0, a, v, u) = (v(i), u(i)).$$

At time $t + 1$. For the left- and right-most cells 1 and n,

$$\begin{array}{rcl} F_1(t+1, a, v, u) & = & f_1(F_1(t, a, v, u), V_2(t, a, v, u), a_1(t)) \\ F_n(t+1, a, v, u) & = & f_n(F_n(t, a, v, u), V_{n-1}(t, a, v, u), a_n(t)) \end{array}$$

and for all other cells $i = 2, \ldots, n-1$,

$$F_i(t+1, a, v, u) = f_i(F_i(t, a, v, u), V_{i-1}(t, a, v, u), V_{i+1}(t, a, v, u), a_i(t)).$$

4 Example SCA models of cardiac tissue

We introduce three SCA models of cardiac tissue that we will use to illustrate our theory and methodology for the remainder of the paper. Each of these models describes electrical behaviour at a different level of biological detail and operates on different time- and space scales. The models are listed below, in descending order of biological detail.

(i) A CODE lattice in which each lattice point comprises a biophysically derived system of ODEs modelling a single cardiac cell, incorporating detailed representations of ionic currents. The coupling between cells in the lattice represents the cell-to-cell junctional conductance.

(ii) A PDE model derived from the FitzHugh-Nagumo equations, designed to reproduce some aspects of propagation of waves of depolarisation in mammalian ventricular tissue including action potential shape and rate-dependent effects (restitution).

(iii) A CA model designed to reproduce the basic properties of excitability and wave-propagation through excitable media in general, not specific to cardiac tissue.

The table below summarises which well-known properties of cardiac tissue are reproduced by each of the three models, where 'accurate' denotes an accurate reproduction of a property, 'approx' denotes a good approximation, 'poor' denotes a crude representation, and '-' denotes that the property is not reproduced by the model at all.

Local temporal properties	**CODE**	**PDE**	**CA**
action potential shape	accurate	approx	poor
restitution	accurate	approx	-
biophysics	approx	-	-
Global spatio-temporal properties			
wave propagation	accurate	approx	poor
vulnerability	accurate	poor	poor
dispersion	accurate	poor	-

The CA model can be described directly as an SCA in the style of Section 3.2 and 3.3. The CODE and PDE models are not algorithmic models and therefore not SCAs. To discover their behaviours however, they must be approximated numerically by algorithms implemented on a computer. There exist many numerical approximation techniques for PDEs and CODEs, most of which are examples of SCAs; the more elaborate methods give efficient approximations to the underlying equations. Because the current paper focuses on theories and applications of abstraction, approximation and multi-level models, rather than on numerical algorithms, we will use a simple (inefficient) numerical integration method—finite differences—and describe these approximations to the CODE and PDE as SCAs in the style of Section 3.2 and 3.3.

The three SCAs are defined in Sections 4.1–4.3. For the ease of illustration, we define only one-dimensional homogeneous versions of the models, and we assume that the input streams supply only electrical stimuli. We fully specify each model by defining its space, cell state and stimulation sets, and its cells' local dynamics; the architecture and equations for the local state functions of each model need not be given, as they are identical in form to those in Section 3.3. We illustrate each of the models using a simulation in which we apply a single stimulation at one end of the strand of tissue, generating a propagating action potential. We discuss the restitution properties of the models in Section 4.4. Parameters of the PDE model are set such that it approximates the action

potential shape, duration and velocity of the CODE model, as well as its restitution properties. Likewise, parameters in the CA model are set to approximate the action potential shape, duration and velocity of the PDE model (the CA does not model restitution properties). Formal notions of approximation of behaviours is the topic of Sections 5 and 6, and this provides a foundation for building hierarchical or multi-level models in Sections 7 and 8.

4.1 A coupled ordinary differential equation model

Consider the model of a single Guinea Pig cardiac cell from the Oxsoft suite [23]. For our purposes it is not necessary to present the (complex) model in full. Of interest here is its basic structure, given by the equations

$$\frac{\mathrm{d}v}{\mathrm{d}t} = G(v, u, e) \quad \frac{\mathrm{d}u}{\mathrm{d}t} = H(v, u, e)$$

where variable $v \in \mathbf{R}$ represents membrane potential, $u \in \mathbf{R}^{16}$ is a vector of 16 ion-concentration and gating variables, $e \in \mathbf{R}$ is external stimulation, and the functions G and H determine the cell's behaviour. A full description of the model is available in [3].

We form a one-dimensional CODE model of a strand of $n > 1$ cardiac cells by *coupling* n copies of the single-cell model. Let $I = \{1, \ldots, n\}$ index the cells in the coupled system, and let $v_i \in \mathbf{R}$, $u_i \in \mathbf{R}^{16}$ and $e_i \in \mathbf{R}$ be voltage, concentration/gating and external stimulation variables respectively for cell $i \in I$. Cell i (for $1 < i < n$) interacts with its immediate left- and right-hand neighbours $i-1$ and $i+1$ (cell 1 interacts only with cell 2, and cell n interacts only with cell $n-1$) in a diffusive fashion with respect to the voltages v_i, v_{i-1} and v_{i+1}. Let the strength of coupling between cells be denoted by a diffusion coefficient $D \in \mathbf{R}$, measured in Siemens [28].

For the membrane potentials of each non-end cell $i = 2, \ldots, n-1$, let

$$\frac{\mathrm{d}v_i}{\mathrm{d}t} = G(v_i, u_i, e_i) + D(v_{i-1} + v_{i+1} - 2v_i),$$

and for cells 1 and n,

$$\frac{\mathrm{d}v_1}{\mathrm{d}t} = G(v_1, u_1, e_1) + D(v_2 - v_1) \quad \text{and} \quad \frac{\mathrm{d}v_n}{\mathrm{d}t} = G(v_n, u_n, e_n) + D(v_{n-1} - v_n).$$

For each $i \in I$, the concentration/gating variables are not coupled and thus:

$$\frac{\mathrm{d}u_i}{\mathrm{d}t} = H(v_i, u_i, e_i).$$

Such coupled ordinary differential equation models have been used in [28] for ventricular tissue, and in [33] for two-dimensional models of sino-atrial node and atrial tissue. We define a simple finite difference approximation to this CODE lattice in the style of Section 3.3.

Tissue state. States of cells in the the SCA CODE model are represented by a pair $(v, u) \in \mathbf{R}^{17}$ where the voltage $v \in \mathbf{R}$ is observable and the concentration/gating state $u \in \mathbf{R}^{16}$ is hidden. Global observable and hidden states of the one-dimensional system are given in the form $v \in \mathbf{R}^I$ and $u \in (\mathbf{R}^{16})^I$ respectively where $v(i) \in \mathbf{R}$ and $u(i) \in \mathbf{R}^{16}$ are the observable and hidden states of cell $i \in I$ respectively.

External stimulation. Each cell $i \in I$ has associated with it a local stream of input stimuli, where stimulation values are from the set $\mathbf{R}$. The model is thus supplied with a set of streams of the form $a \in [T \to \mathbf{R}]^I$.

Local dynamics. Each non-end cell $i = 2, \ldots, n-1$ has local dynamics $f_i = (g_i, h_i)$ where

$$\begin{aligned} g_i &: \mathbf{R}^{17} \times \mathbf{R} \times \mathbf{R} \times \mathbf{R} \to \mathbf{R} \\ h_i &: \mathbf{R}^{17} \times \mathbf{R} \times \mathbf{R} \times \mathbf{R} \to \mathbf{R}^{16} \end{aligned}$$

give the next observable and hidden states of i respectively. The functions g_i and h_i are defined, for all cell states $(v, u) \in \mathbf{R}^{17}$, left and right neighbour's membrane potentials $v_l, v_r \in R$, and input stimuli values $a \in R$, by

$$\begin{aligned} g_i(v, u, v_l, v_r, a) &= v + \Delta t(G(v, u, a) + D(v_l + v_r - 2v)) \\ h_i(v, u, v_l, v_r, a) &= u + \Delta t(H(v, u, a)), \end{aligned}$$

where Δt is a numerical time-step parameter. The left- and right-most cells $i = 1, n$ have local dynamics $f_i = (g_i, h_i)$ where

$$\begin{aligned} g_i &: \mathbf{R}^{17} \times \mathbf{R} \times \mathbf{R} \to \mathbf{R} \\ h_i &: \mathbf{R}^{17} \times \mathbf{R} \times \mathbf{R} \to \mathbf{R}^{16} \end{aligned}$$

are defined, for all states $(v, u) \in \mathbf{R}^{17}$, left or right neighbour's membrane potentials $v_{l/r} \in \mathbf{R}$ and input stimuli values $a \in \mathbf{R}$, by

$$\begin{aligned} g_i(v, u, v_{l/r}, a) &= v + \Delta t(G(v, u, a) + D(v_{l/r} - v)) \\ h_i(v, u, v_{l/r}, a) &= u + \Delta t(H(v, u, a)). \end{aligned}$$

Behaviour. To demonstrate the behaviour of the Oxsoft CODE model, consider a lattice of $n = 2000$ coupled cells reconstructing a 160mm strand of tissue (taking 80μm as the length of a cardiac cell). This very long strand is used so we can illustrate the spatial distribution of potential in a travelling wave.

For numerical stability, we set the numerical timestep $\Delta t = 0.01$ms. An action potential conduction velocity of 0.6ms^{-1}, which has been observed by [28], is achieved by setting parameter value $D = 33875$, representing a cell–cell coupling of $3.4\mu S$. Figure 6 gives snapshots of the global observable state showing an action potential propagating along the model at 50ms (5000 clock-cycle) intervals following a single 30nA stimulation of the 8 left-most cells for an initial period of 2.15ms (given by input stimuli streams $a_i(t) = -30$ for $1 \leq i \leq 8$ and $t < 215$, $a_i(t) = 0$ otherwise), given that the model begins in a uniformly resting state (given by appropriate values of v and u). This choice of stimulation strength and duration ensures that an action potential is generated for this (single stimulation) case and for periodic stimulation (see Section 4.4); and it allows us to compare this model with the other two models; see Section 6. Figure 7 shows a trace of the voltage, i_{Ca}, i_{Na} and i_K currents of cell 1000 against time for the same simulation.

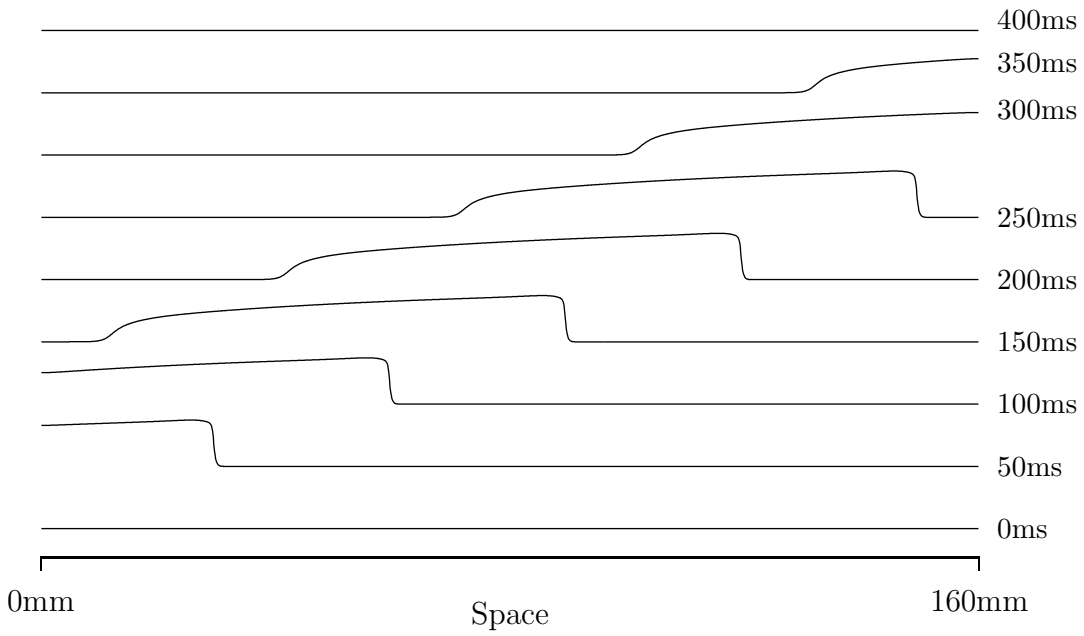

Figure 6: An action potential travelling from left-to-right along the Oxsoft CODE model following a single stimulus at the 8 left-most cells.

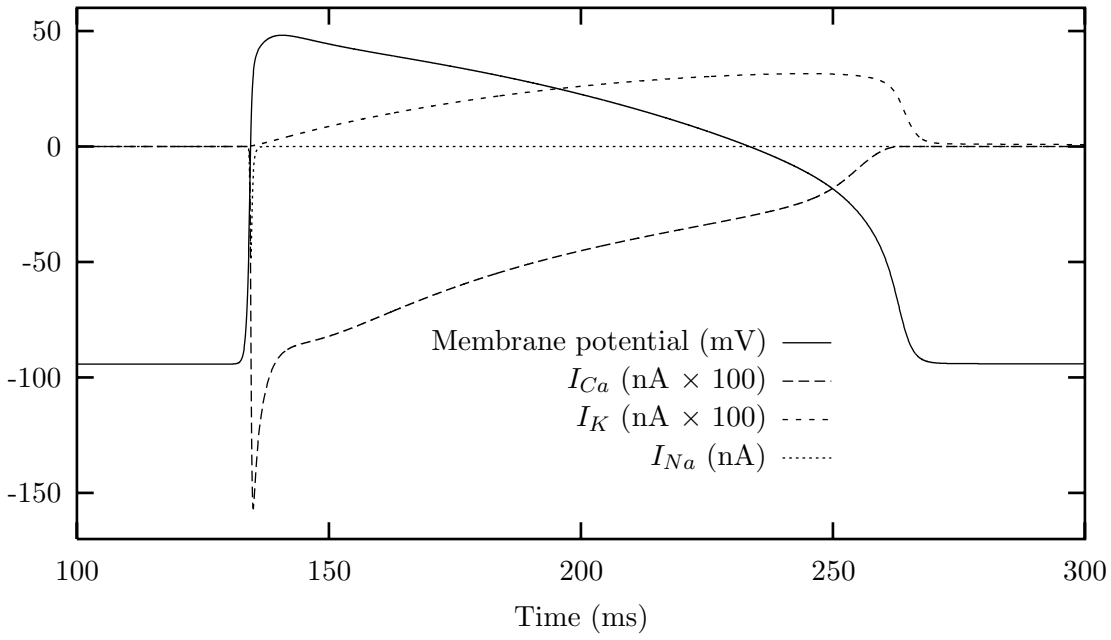

Figure 7: The membrane potential and i_{Ca}, i_{Na} and i_K currents of cell 1000 during the period 100ms–300ms of the simulation of Figure 6.

4.2 A partial differential equation model

We now describe a two-variable partial differential equation model of cardiac excitation developed by Aliev and Panfilov [1]. The model is a modification of the FitzHugh-Nagumo equations [7], intended to reproduce acceptable action potential and restitution properties of cardiac tissue. The model consists of two equations, one describing fast (excitation or voltage) processes, the other slow (recovery) processes:

$$\begin{aligned}\frac{\partial v}{\partial t} &= -8v(v-0.1)(v-1) - vu + e + \frac{\partial^2 v}{\partial x^2} \\ \frac{\partial u}{\partial t} &= \varepsilon(v,u)(-u - 8v(v-1.1)).\end{aligned}$$

Here, t is time, x is space, e is a time- and space-dependent variable that models electrical stimulation, $\varepsilon(v,u) = 0.002 + \mu_1 u/(v+\mu_2)$ represents the excitability of the tissue, and $\mu_1, \mu_2 \in \mathbf{R}$ are parameters We define a finite difference approximation to this model in the style of Section 3.3.

Tissue state. States of a cell are represented by pairs of the form $(v,u) \in \mathbf{R}^2$, where the voltage v is observable and the recovery state u is hidden. Global observable and hidden states of the one-dimensional system are thus represented in the form $v \in \mathbf{R}^I$ and $u \in \mathbf{R}^I$ respectively.

External stimulation. Each cell $i \in I$ may be stimulated by a stream of external inputs from the set $\mathbf{R}$. The model is thus supplied with a set of streams of the form $a \in [T \to \mathbf{R}]^I$.

Local dynamics. Each non-end cell $i = 2, \ldots, n-1$ has local dynamics

$$f_i = (g_i, h_i) : \mathbf{R}^2 \times \mathbf{R} \times \mathbf{R} \times \mathbf{R} \to \mathbf{R}^2$$

where g_i and h_i give the next observable and hidden states of i and are defined, for all cell states $(v,u) \in \mathbf{R}^2$, left and right neighbour's voltages $v_l, v_r \in \mathbf{R}$, and input stimuli $a \in \mathbf{R}$, by

$$\begin{aligned}g_i(v,u,v_l,v_r,a) &= v + \Delta t(-8v(v-0.1)(v-1) - vu + a) + \frac{\Delta t}{\Delta x^2}(v_l - 2v + v_r) \\ h_i(v,u,v_l,v_r,a) &= u + \Delta t(\varepsilon(v,u)(-u - 8v(v-1.1))).\end{aligned}$$

where Δt (timestep) and Δx (space step) are numerical parameters. To give non-flux boundary conditions, the left- and right-most cells $i = 1, n$ share local dynamics

$$f_i = (g_i, h_i) : \mathbf{R}^2 \times \mathbf{R} \times \mathbf{R} \to \mathbf{R}^2$$

where g_i and h_i are defined, for all cell states $(v,u) \in \mathbf{R}$, neighbour's voltages $v_{l/r} \in \mathbf{R}$, and input stimuli $a \in \mathbf{R}$, by

$$\begin{aligned}g_i(v,u,v_{l/r},a) &= v + \Delta t(-8v(v-0.1)(v-1) - vu + a) + \frac{\Delta t}{\Delta x^2}(v_{l/r} - v) \\ h_i(v,u,v_{l/r},a) &= u + \Delta t(\varepsilon(v,u)(-u - 8v(v-1.1))).\end{aligned}$$

Behaviour. We illustrate Aliev and Panfilov's model using a simulation corresponding with that of the Oxsoft CODE model in Section 4.1. Consider a system comprising $n = 500$ cells representing a

160mm length of cardiac tissue as for the Oxsoft CODE model, thus assuming a PDE cell represents a 0.32mm strand of tissue or 4 cardiac/Oxsoft cells. To give a numerically stable solution to the PDE, to approximate the action potential shape, duration and conduction, and the restitution properties of the Oxsoft model, we set parameter values $\mu_1 = 0.01$, $\mu_2 = 0.14$, numerical time-step $\Delta t = 0.1124$ (representing 0.05ms or 5 clock cycles of the Oxsoft model) and space-step $\Delta x = 1.57$. Figure 8 shows the potential along the system at 50ms intervals following an initial stimulus at the two left-most cells with value 0.12 for a period of 43 clock cycles or 2.15ms (i.e. taking $a_1(t) = a_2(t) = 0.12$ for $t < 43$ and $a_i(t) = 0$ for all other i and t) for an initially resting system ($v(i) = u(i) = 0$ for all i). The choice of stimulation strength and duration is derived from the formal comparison of this and the other two models; see Section 6.

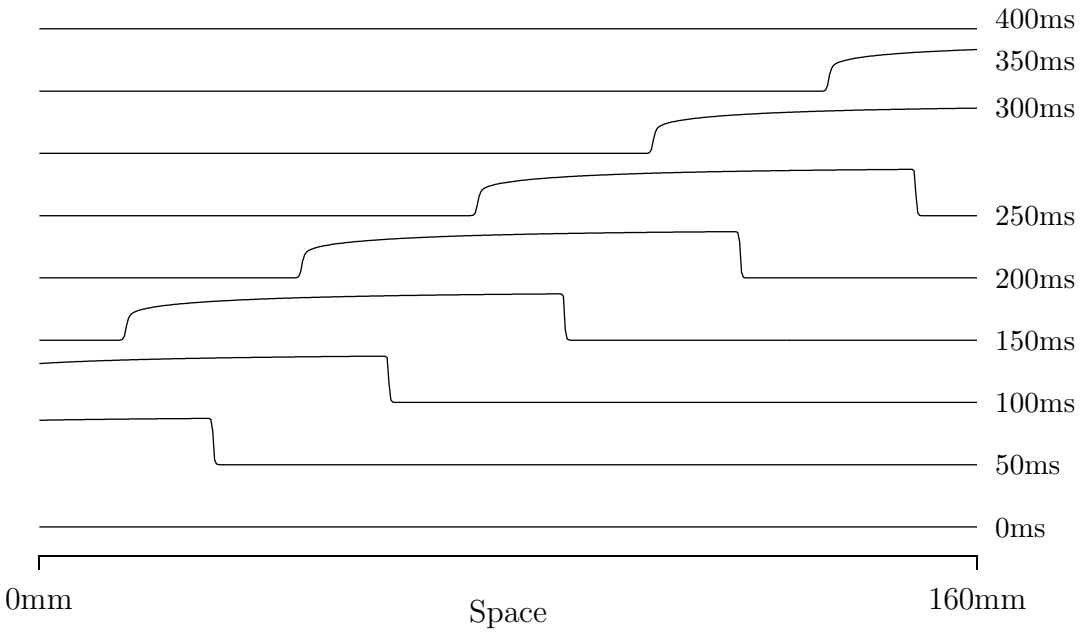

Figure 8: An action potential propagating from left-to-right along Aliev and Panfilov's PDE model following a single stimulus at the two left-most cells.

4.3 A cellular automaton model

As a final model, consider the cellular automaton introduced by Gerhardt, Schuster and Tyson in [8], here simplified from the original two-dimensional form. This model reproduces the basic properties of excitable media including wave propagation and annihilation on collision, but does not attempt to give a realistic cardiac action potential, dispersion effects, or cardiac restitution properties. We describe the model directly as an SCA in the style of Sections 3.2 and 3.3.

Tissue state. Let $S = A \times B$ be the set of all possible cell states, where $A = \{0, 1\}$ and $B = \{0, 1, \ldots, B_{max}\}$ are observable cell excitation and hidden recovery sets respectively. The state of a cell is represented by a pair of the form $(v, u) \in S$ where $(0, 0)$ represents resting tissue, $(1, u)$ for

any u represents excited tissue, and $(0, u)$ for $u \neq 0$ represents refractory tissue.

External stimulation. Let $P = \{0, 1\}$ be a set of all local external stimulation values, where 1 represents the presence of stimulation and 0 represents the absence of stimulation. The model is thus supplied with a set of streams of the form $a \in [T \to \{0, 1\}]^I$.

Local dynamics. Let the local dynamics $f_i = (g_i, h_i) : S \times A^2 \times P \to S$ of all non-end cells $i = 2, \ldots, n-1$ be defined, for all cell states $(v, u) \in S$, left and right neighbour's excitation states $v_l, v_r \in A$ and local stimulation values $a \in P$, as follows. First, g_i is defined by

$$g_i(v, u, v_l, v_r, a) = \begin{cases} 1 & \text{if } v, u = 0 \text{ and } v_l + v_r + a \geq 1 \\ 0 & \text{if } v = 1 \text{ and } u = B_{max} \\ u & \text{otherwise.} \end{cases}$$

Informally, this means that a resting cell becomes excited if either of its neighbours are excited, or if it is stimulated externally; an excited cell becomes unexcited if its recovery state has reached B_{max}; otherwise the excitation state of a cell remains unaltered. Second, h_i is defined by

$$h_i(v, u, v_l, v_r, a) = \begin{cases} min(u + up, B_{max}) & \text{if } v = 1 \\ max(u - down, 0) & \text{if } v = 0 \end{cases}$$

where $up, down > 0$ are parameters. Informally, in an excited state, a cell's recovery state increases to a maximum value of B_{max} and, in an unexcited state, it decreases to a minimum value of 0; the parameters *up* and *down* determine the length of time a cell remains excited and refractory respectively.

Similarly, for left- and right-most cells $i = 1, n$, we define $f_i = (g_i, h_i) : S \times A \times P \to S$, for all current cell states $(v, u) \in S$, neighbour's excitation states $v_{l/r} \in A$ and local stimulation values $a \in P$, by:

$$g_i(v, u, v_{l/r}, a) = \begin{cases} 1 & \text{if } v, u = 0 \text{ and } v_{l/r} + a \geq 1 \\ 0 & \text{if } v = 1 \; u = B_{max} \\ u & \text{otherwise} \end{cases}$$

$$h_i(v, u, v_{l/r}, a) = \begin{cases} min(u + up, B_{max}) & \text{if } v = 1 \\ max(u - down, 0) & \text{if } v = 0. \end{cases}$$

Behaviour. We illustrate the behaviour of the CA model using a simulation corresponding with those of the CODE and PDE models in Sections 4.1 and 4.2. We use an $n = 250$ cell model to represent a 160mm strand of tissue as for the CODE and PDE models; each CA cell thus represents a 0.64mm length of 8 cardiac (or Oxsoft cells) or 2 PDE cells.

We wish the CA to reproduce the basic action potential duration and velocity properties of the other, more detailed, models. Observe from the definition of the CA model, that an unhindered action potential propagates at the speed of one CA cell per timestep. To achieve a propagation velocity of 0.6ms^{-1} we take each timestep to represent approx 1.065ms, which corresponds with 21.3 PDE clock-cycles. We approximate the action potential duration properties of the PDE model by setting the parameter values $B_{max} = 116$, $up = 1$ and $down = 24$. (These particular parameter values not only approximate the action potential duration and velocity properties of the PDE model, but

also its vulnerability properties; this, however, is not of primary concern here). Figure 9 illustrates a propagating wave in the CA model following a single stimulus at the left-most cell for two clock cycles or 2.13ms (given by $a_1(0) = a_1(1) = 1$ and $a_i(t) = 0$ for all other i and t).

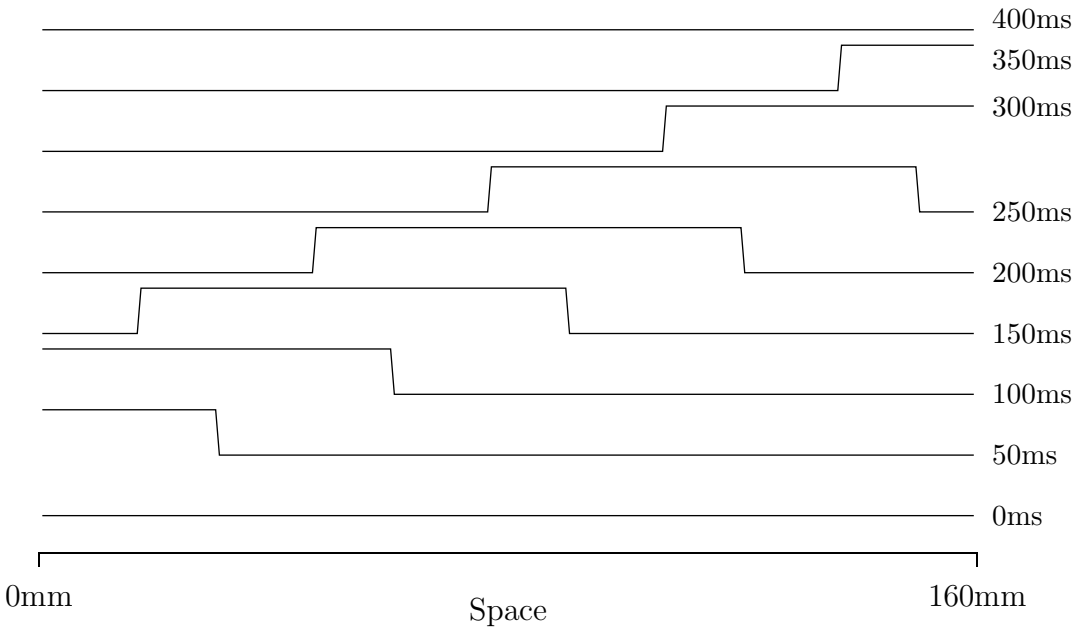

Figure 9: An action potential travelling from left-to-right along Gerhardt, Schuster and Tyson's CA model following a single stimulus at the left-most cell.

4.4 Models' restitution properties

In ventricular cells, as a result of the different timescales for the membrane currents and intracellular ionic concentrations the shape of the action potential changes with time since the preceeding action potential. This dependence can be simply represented for periodic activity by an electrical restitution curve—the duration of the action potential as a function of period. The action potential duration influences the strength of contraction, and so the electrical restitution properties give rise to mechanical restitution properties. Rate dependent changes are important normal cardiac activity, are severely altered during cardiac infarction, and have been implicated in the development of spatio-temporal irregularity in cardiac tissue [24]. Thus a good approximation of cardiac electrical activity needs to retain its restitution properties. Figure 10 illustrates the restitution properties of the three models. The results were obtained by stimulating at the left of each system with the period of stimulation shown, and obtaining action potential durations at a cell 90% along from the left of the model. Action potentials durations were measured at -40mV for the Oxsoft model, and at corresponding values of 0.38 and 1 (see Section 6) for the PDE and CA models.

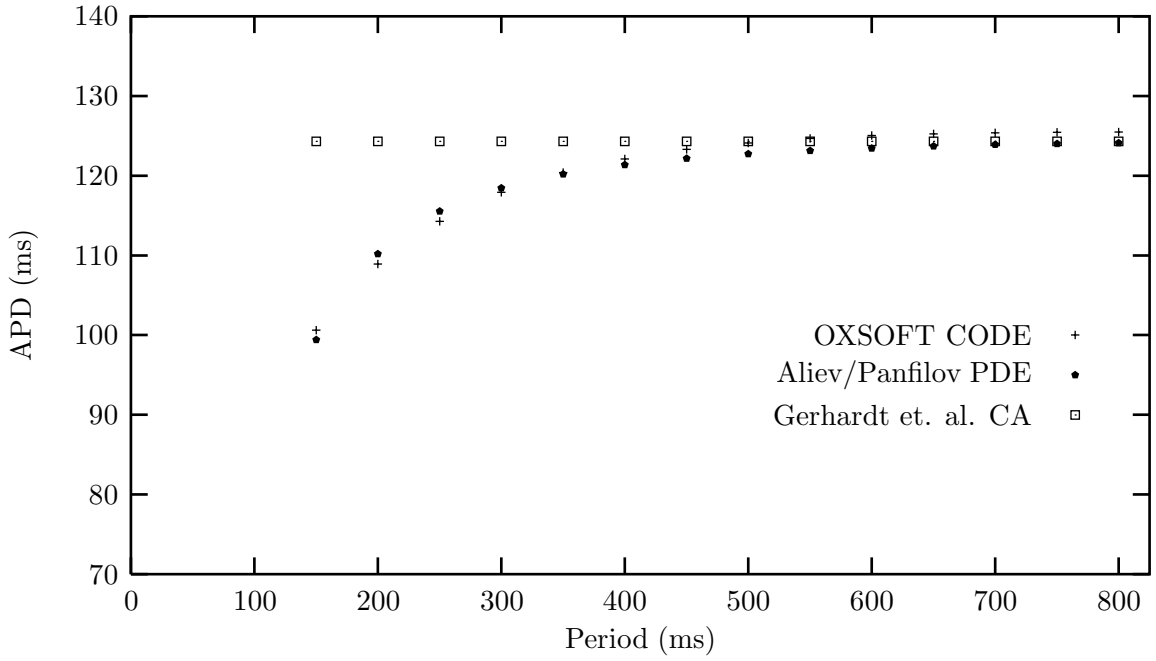

Figure 10: Restitution properties of the three models. Over maintained stimulation periods from 150ms to 800ms the action potential durations of the CODE and PDE models are within 5%. The CA model does not incorporate rate-dependent changes and therefore action potential durations are constant over all stimulation rates.

5 Abstraction and approximation between models

The one-dimensional CODE, PDE and CA models of Section 4 all support travelling waves and wave trains, which we wish to be able to compare and match. The PDE and CODE models both exhibit rate-dependent changes in action potential duration, as illustrated in Figure 10, and so the PDE can be considered to approximate the solitary wave, wave train and action potential restitution properties of the CODE model. The CA can be considered to approximate the propagation velocity and action potential duration of solitary waves of the PDE (and CODE) model. If these different models are to be used to simulate the same behaviours, or to be coupled into hybrid multi-level models, we need to be able to formalise these notions of approximation.

Consider two models SCA M_1 and M_2 specified by

$$\begin{aligned} M_1 &= (I_1, p_1, r_1, T_1, A_1, B_1, P_1, \langle f_{1,i} \mid i \in I_1 \rangle) \\ M_2 &= (I_2, p_2, r_2, T_2, A_2, B_2, P_2, \langle f_{2,i} \mid i \in I_2 \rangle) \end{aligned}$$

respectively. We compare the *observable behaviours* of these two models and define formally the notion that the observable behaviour of M_2 *approximates* the observable behaviour of M_1. We first compare formally the *components* of M_1 and M_2 in Section 5.1. Then, in Section 5.2, we present a formal definition of approximation of observable behaviour between the two models, with respect

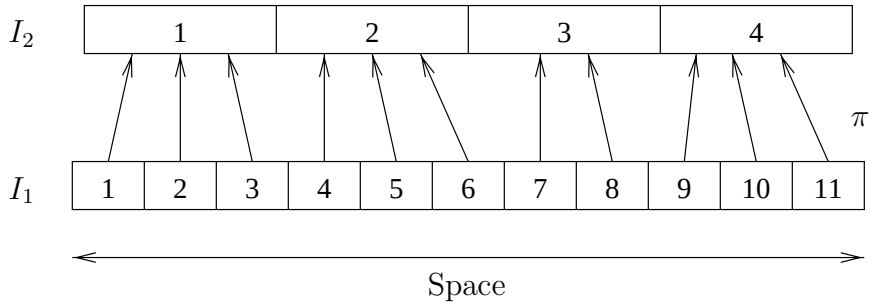

Figure 11: An example one-dimensional space abstraction map π.

to the component abstractions. In Section 6, we compare the components and behaviours of the CODE, PDE and CA models of Section 4.

5.1 Abstractions between model components

In order to compare the behaviours of M_1 and M_2 we first compare the space sets I_1 and I_2, the clocks T_1 and T_2, global observable and hidden state sets $A_1^{I_1}$, $A_2^{I_2}$, $B_1^{I_1}$ and $B_2^{I_2}$, and global parameter stream sets $[T_1 \to P_1]^{I_1}$ and $[T_2 \to P_2]^{I_2}$. We begin by comparing the sets I_1 and I_2 of points or cells.

Spaces. We shall assume that each cell $j \in I_2$ of model M_2 abstracts a non-empty set of cells of I_2 and that each cell $i \in I_1$ of model M_1 is abstracted by a single cell in I_2 . A *space abstraction map*

$$\pi : I_1 \to I_2$$

is thus a surjective function, where each M_1 cell $i \in I_1$ is abstracted in M_2 by cell $\pi(i) \in I_2$. The inverse $\pi^{-1} : I_2 \to Powerset(I_1)$ of π is defined, for each M_2 cell $j \in I_2$, by

$$\pi^{-1}(j) = \{i \in I_2 \mid \pi(i) = j\}$$

where $\pi^{-1}(j)$ is termed the *subspace* of all M_1 cells abstracted by cell j.

Consider, for example, two one-dimensional spaces $I_1 = \{1, 2, \ldots, 11\}$ and $I_2 = \{1, 2, 3, 4\}$, related with a (typical) abstraction map π, illustrated in Figure 11. Here $\pi(4) = 2$, $\pi(5) = 2$, $\pi(6) = 2$ and $\pi^{-1}(2) = \{4, 5, 6\}$. It is common in the case of one-dimensional models for π to be monotonic, such that each cell $j \in I_2$ abstracts a *strand* of one or more neighbouring cells in I_1 (often, these strands will contain a similar number of cells for each $j \in I_2$).

Clocks. We now consider the models' global clocks T_1 and T_2. A *clock abstraction map* or *retiming*

$$\lambda : T_1 \to T_2$$

is a surjective, monotonic function, with the intention that clock cycle $t \in T_1$ is abstracted by clock cycle $\lambda(t) \in T_2$. From a retiming λ, we determine an *immersion* $\overline{\lambda} : T_2 \to T_1$ defined, for all $t \in T_2$, by

$$\overline{\lambda}(t) = \text{ min } t' \in T_1 \text{ such that } \lambda(t') = t$$

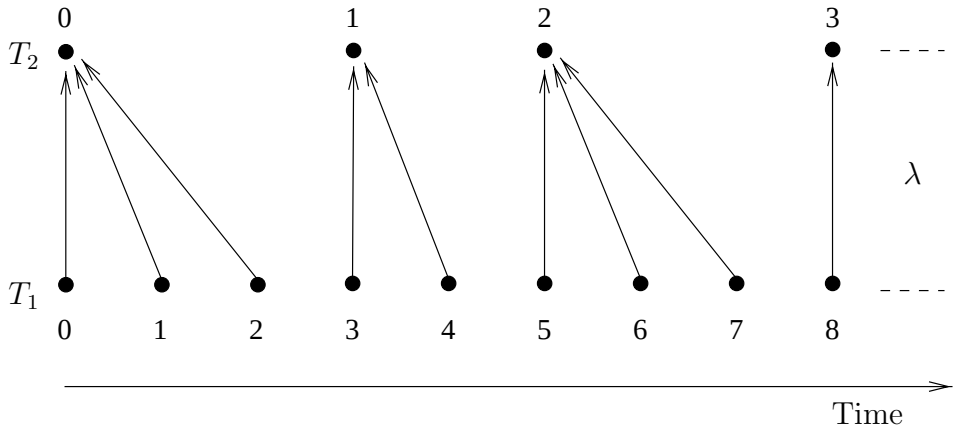

Figure 12: The linear retiming $\lambda(t) = \lfloor \frac{t}{2.5} \rfloor$.

where, for $t \in T_2$, $\overline{\lambda}(t)$ is the *first* clock cycle in T_1 abstracted by t. Let the range of $\overline{\lambda}$, comprising all such first clock cycles, be denoted $Start_\lambda \subseteq T_1$. For models of cardiac tissue, we are usually interested in expressing the idea that clock T_1 is r times faster than clock T_2 for some $r \in \mathbf{R}$ where $r \geq 1$; this is accomplished by means of a *linear retiming* of the form

$$\lambda(t) = \left\lfloor \frac{t}{r} \right\rfloor .$$

The linear-retiming for the case $r = 2.5$ is illustrated in Figure 12. Here the immersion $\overline{\lambda} : T_2 \to T_1$ is given by $\overline{\lambda}(0) = 0$, $\overline{\lambda}(1) = 3$, $\overline{\lambda}(2) = 5, \ldots$ and the set $Start_\lambda$ is given by $Start_\lambda = \{0, 3, 5, \ldots\}$.

Global observable states. Global observable states of M_1 are compared with those of M_2 by means of an *observable state abstraction map*

$$\phi : A_1^{I_1} \to A_2^{I_2}$$

with the intention that an observable state $v \in A_1^{I_1}$ of model M_1 is abstracted or approximated by state $\phi(v) \in A_2^{I_2}$ of model M_2. The global map ϕ is determined by *local* abstraction mappings

$$\phi_j : A_1^{\pi^{-1}(j)} \to A_2$$

for each cell $j \in I_2$, with the intention that a local observable state $v_l \in A_1^{\pi^{-1}(j)}$ (i.e. a state of the subspace $\pi^{-1}(j) \subseteq I_1$ that j abstracts) is abstracted or approximated by observable state $\phi_j(v_l) \in A_2$ of cell j. (Often, where the observable state sets A_1 and A_2 are (subsets) of $\mathbf{R}$, ϕ_j amounts to scaling the average value of the observable states of the cells in the subspace $\pi^{-1}(j)$. It is also usual for the maps ϕ_j to be uniform throughout I_2.)

The global map $\phi : A_1^{I_1} \to A_2^{I_2}$ is defined, for all global observable states $v \in A_1^{I_1}$ of M_1 and for all M_2 cells $j \in I_2$, by

$$\phi(v)(j) = \phi_j(v|_{\pi^{-1}(j)})$$

where $v|_{\pi^{-1}(j)} \in A_1^{\pi^{-1}(j)}$ is the state of the subspace $\pi^{-1}(j)$ for global state v.

Global hidden states. Global hidden states of M_1 are compared with those of M_2 by means of a *hidden state abstraction map*

$$\psi : B_1^{I_1} \to B_2^{I_2}$$

with the intention that a global hidden state $u \in A_1^{I_1}$ of model M_1 is abstracted or approximated in model M_2 by $\psi(u) \in B_2^{I_2}$. As with observable states, the global map ψ is determined from *local maps*

$$\psi_j : B_1^{\pi^{-1}(j)} \to B_2$$

for each cell $j \in I_2$ that relate a local hidden state $u \in B_1^{\pi^{-1}(j)}$ of subspace $\pi^{-1}(j)$ with state of $\psi_j(u) \in B_2$ of cell j. Unlike observable states however, due to the complexity of the set B_1 for many models (for example, the Oxsoft model of Section 4.1) not all possible local hidden states of $\pi^{-1}(j)$ can realistically be related to hidden states of cell j (the vast state-space of the 16 ionic current/gating variables of the Oxsoft model contain states that cannot be properly approximated by the single recovery variable of Aliev and Panfilov's PDE model). Each ψ_j is therefore, in general, a partial function, which means that the global hidden states, and hence global hidden behaviours cannot be fully compared. However, global *observable* behaviours of models can be compared in the case of partial hidden state abstraction maps; we require only that the domains $dom(\psi_j)$ for all $j \in I_2$ are non-empty. Typically, $dom(\psi_j)$ will comprise values that represent resting tissue for M_1 that are often easily mapped onto corresponding rest states in M_2.

The global map $\psi : B_1^{I_1} \to B_2^{I_2}$ is thus in general also a partial function with domain $dom(\psi) = \{u \in B_1^{I_1} \mid u|_{\pi^{-1}(j)} \in dom(\psi_j) \text{ for all } j \in I_2\}$ and is defined, for all global hidden states $u \in dom(\psi)$ of M_1 and M_2 cells $j \in I_2$, by

$$\psi(u)(j) = \psi_j(u|_{\pi^{-1}(j)}).$$

Parameter streams. Parameter stream sets for model M_1 are compared with those for M_2 using a *parameter stream abstraction map*

$$\Theta : [T_1 \to P_1]^{I_1} \to [T_2 \to P_2]^{I_2}$$

with the intention that a set $a \in [T_1 \to P_1]^{I_1}$ of parameter streams for model M_1 is abstracted or approximated in model M_2 by $\Theta(a) \in [T_2 \to P_2]^{I_2}$. This global map is determined from *local parameter sequence maps* which are defined with respect to the space abstraction map π and retiming λ in the following way.

Consider first a single stream $a \in [T_1 \to P_1]$ appropriate for any M_1 cell. For any M_2 clock cycle $t \in T_2$, let *cell-seq*(a, t) denote the sequence

$$(a(\overline{\lambda}(t)), a(\overline{\lambda}(t)+1), \ldots, a(\overline{\lambda}(t+1)-1)) \in P_1^*$$

of parameters supplied by stream a for the sequence $(\overline{\lambda}(t), \overline{\lambda}(t)+1, \ldots, \overline{\lambda}(t+1)-1) \in T_1^*$ of M_1 clock cycles abstracted by t. For example, for the retiming illustrated in Figure 12, *cell-seq*$(a, 0) = (a(0), a(1), a(2)) \in P_1^*$ and *cell-seq*$(1) = (a(3), a(4)) \in P_1^*$.

Consider now a set $a = (a_i \mid i \in \pi^{-1}(j)) \in [T_1 \to P_1]^{\pi^{-1}(j)}$ of parameter streams supplying the subspace $\pi^{-1}(j)$ of M_1 cells abstracted by an M_2 cell $j \in I_2$. For any clock cycle $t \in T_2$, let

subspace-seq(a,t) denote the set

$$(\textit{cell-seq}(a_i,t) \mid i \in \pi^{-1}(j)) \in (P_1^*)^{\pi^{-1}(j)}$$

of parameter sequences supplied by a to subspace $\pi^{-1}(j)$ for the sequence of M_1 clock cycles abstracted by $t \in T_2$. For example, for the space abstraction map π illustrated in Figure 11 and the retiming λ of Figure 12, let $a \in [T_1 \to P_1]^{\{4,5,6\}}$ be a set of streams that are appropriate input for the subspace $\pi^{-1}(j) = \{4,5,6\}$ abstracted by j. Then *subspace-seq*$(a,1) = ((a_i(3), a_i(4)) \mid i \in \{4,5,6\}) \in (P_1^*)^{\{4,5,6\}}$ comprising 6 parameter values.

A *local parameter sequence abstraction map* for cell $j \in I_2$ takes the form

$$\theta_j : (P_1^*)^{\pi^{-1}(j)} \to P_2$$

with the intention that a parameter sequence set $b \in (P_1^*)^{\pi^{-1}(j)}$ is abstracted or approximated by the single parameter value $\theta_j(b) \in P_2$ for cell j. As with hidden state abstraction, due to the complexity of the set $[T_1 \to P_1]$ for some models, we allow each θ_j to be partial, but we require that its domain $dom(\theta_j) \subseteq (P_1^*)^{\pi^{-1}(j)}$ is non-empty.

The global stream map $\Theta : [T_1 \to P_1]^{I_1} \to [T_2 \to P_2]^{I_2}$, determined for the local sequence maps θ_j for $j \in J$, has domain

$$dom(\Theta) = \{a \in [T_1 \to P_1]^{I_1} \mid \textit{subspace-seq}(a|_{\pi^{-1}(j)}, t) \in dom(\theta_j) \text{ for all } j \in I_2 \text{ and } t \in T_2\}$$

and is defined, for all $a \in [T_1 \to P_1]^{I_1}$, $j \in I_2$ and $t \in T_2$, by

$$\Theta(a)(j)(t) = \theta_j(\textit{subspace-seq}(a|_{\pi^{-1}(j)}, t)).$$

Abstraction tuples. The abstraction maps are conveniently collected together into an *abstraction tuple*

$$\Gamma = (\pi, \lambda, \phi, \psi, \Theta)$$

comprising space, time, global observable state, global hidden state and global parameter stream abstraction maps. As we have seen, the global observable and hidden state maps ϕ and ψ are determined by local maps ϕ_j and ψ_j, and the parameter stream abstraction map Θ is determined by local parameter sequence maps θ_j. We may therefore alternatively collect together the retiming, space abstraction and the families of local maps for to give a *local abstraction tuple*

$$\gamma = (\pi, \lambda, \langle \phi_j, \psi_j, \theta_j \mid j \in I_2 \rangle)$$

which completely determines Γ.

5.2 Approximation of models' observable behaviours

The component abstraction mappings of the previous section provide a basis for comparing the global observable spatio-temporal behaviour of two models M_1 and M_2. There are many ways in which behaviours can be compared; for example, by comparison of the positions of wavefronts and/or wave

backs through time under initial conditions and input streams that correspond with respect to the above abstraction maps. Here we provide one simple notion of approximation which compares the excitation or voltage values across the spaces I_1 and I_2 of M_1 and M_2, given respectively by the global observable state functions

$$\begin{aligned} V_1 &: T_1 \times [T_1 \to P_1^{I_1}] \times A_1^{I_1} \times B_1^{I_1} \to A^{I_1} \\ V_2 &: T_2 \times [T_2 \to P_2^{I_2}] \times A_2^{I_2} \times B_2^{I_2} \to A^{I_2}. \end{aligned}$$

Let

$$d : A_2^{I_2} \times A_2^{I_2} \to \mathbf{R}^+$$

be any metric that measures the *distance* between two global observable states of M_2. In the case that A_2 is a subset of the reals or integers, d might be defined for all pairs $v_1, v_2 \in A_2^{I_2}$ of observable states, by

$$d(v_1, v_2) = \sum_{j \in I_2} \mid v_1(j) - v_2(j) \mid$$

to give the sum of the differences of the local states $v_1(j)$ and $v_2(j)$ for each cell $j \in I_2$.

We use d to compare the following two global observable states where $t \in Start_\lambda$ (a clock cycle that is the first abstracted by a T_2 clock cycle), initial observable state $v \in A_1^{I_1}$, hidden state $u \in dom(\psi)$ and parameter stream set $a \in dom(\Theta)$:

(i) the state $\phi(V_1(t, a, v, u))$ abstracted from the state $V_1(t, a, v, u)$ of M_1 at time t, given initial states v and u and parameter stream set a; and

(ii) the state $V_2(\lambda(t), \Theta(a), \phi(v), \psi(u))$ of model M_2 at time $\lambda(t) \in T_2$ abstracting t, given abstractions $\phi(v)$, $\psi(u)$ and $\Theta(a)$ of the initial states v and u, and parameter stream set a.

We say that the global observable behaviour of M_2 approximates that of M_1 if the value given by d for these states is less than some chosen tolerance $\varepsilon \in \mathbf{R}^+$, for all values of t, v, u and a. Formally, we have the following definition.

Definition. Model M_2 approximates M_1 with respect to abstraction tuple $\Gamma = (\pi, \lambda, \phi, \psi, \Theta)$, and tolerance ε, if for all $t \in Start_\lambda$, $a \in dom(\Theta)$ $v \in A_1^{I_1}$ and $u \in dom(\psi)$,

$$d(\phi(V_1(t, a, v, u)), V_2(\lambda(t), \Theta(a), \phi(v), \psi(u))) < \varepsilon.$$

Thus any notion that the global observable behaviour of M_2 approximates that of M_1 with respect to Γ is determined by a metric d and a tolerance ε which we may collect together into an *approximation tuple*

$$\Omega = (d, \varepsilon).$$

This framework for approximation and abstraction allows the formal construction of hierarchies of models. Figure Figure 13b illustrates a hierarchy of three models and can be viewed as a formalisation of Figure 2.

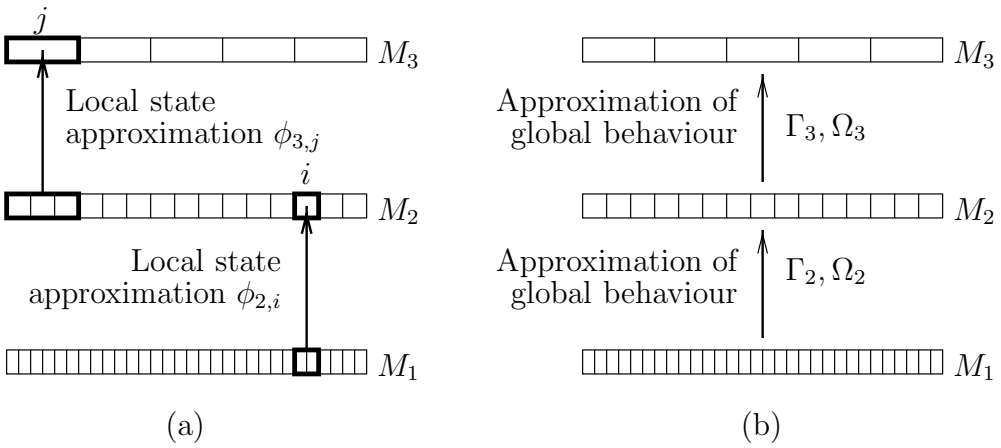

Figure 13: The hierarchy of models M_1, M_2 and M_3.(a) The abstractions tuples Γ_2 and Γ_3 comparing the components of the PDE mode M_2 with the CODE model M_1, and the CA model M_3 with the PDE model M_2, are and defined in terms of comparisons of local abstraction tuples γ_2 and γ_3. (b) The notion that the observable behaviour of the M_2 approximates that of the M_1 is given by approximation tuple Ω_2 and component abstraction tuple Γ_2. Similarly, the notion that the behaviour of M_3 approximates that M_2 is given by Ω_3 and γ_3.

In practice, it is unlikely that M_2 will approximate the observable behaviour of M_1 for all possible clock cycles, initial states and parameter stream sets even for reasonably large values of ε. In two whole-heart models, for example, given an initial state and parameter streams that initiate some complex arrhythmic behaviour, the areas of excitation are extremely unlikely to match closely at each time cycle after the onset of the arrhythmia, although the global observable states may be qualitatively similar. We may therefore define a weaker notion of approximation where observable states are compared only for subsets of clock cycles, initial states and parameters. Let $sub(T_1) \subseteq Start_\lambda$, $sub([T_1 \to P_1]^{I_1}) \subseteq dom(\Theta)$, $sub(A_1^{I_1}) \subseteq A_1^{I_1}$ and $sub(B_1^{I_1}) \subseteq dom(\psi)$ be any chosen subsets. Then M_2 is an approximation of M_1 with respect to abstraction tuple Γ, the above subsets, and tolerance ε, if for all $t \in sub(T_1)$, $a \in sub([T_1 \to P_1]^{I_1})$, $v \in sub(A_1^{I_1})$ and $u \in sub(B_1^{I_1})$, the above inequality holds. An approximation tuple for such a case is of the form

$$\Omega = (d, \varepsilon, sub(T_1), sub([T_1 \to P_1]^{I_1}), sub(A_1^{I_1}), sub(B_1^{I_1})).$$

A fact important for practical purposes, is that when all the subsets are finite, approximation can be exhaustively tested for.

We may also wish to restrict our interest to comparing the behaviour of M_1 with the abstracted behaviour of M_2 only within some subspace $J \subseteq I_2$; for example, with whole-heart models, we may choose to concentrate on the behaviour within the ventricles. Such subspace restrictions can be achieved simply by a redefinition of the state-comparison metric d; for example, for all pairs $v_1, v_2 \in A_2^{I_2}$ of observable states, let

$$d(v_1, v_2) = \sum_{j \in J} \mid v_1(j) - v_2(j) \mid .$$

6 Examples of abstractions and approximations

Here we apply the notions of abstraction and approximation developed in Section 5 to the three one-dimensional SCA models of tissue from Section 4. Let M_1 be the Oxsoft CODE model of Section 4.1, let M_2 be Aliev and Panfilov's PDE model of Section 4.2 and let M_3 be Gerhardt, Schuster and Tyson's CA model of Section 4.3; to avoid confusion we rename the component given in Section 4 such that the models are specified by:

$$\begin{aligned} M_1 &= (I_1, p_1, r_1, T_1, A_1, B_1, P_1, \langle f_{1,i} \mid i \in I_1 \rangle) \\ M_2 &= (I_2, p_2, r_2, T_2, A_2, B_2, P_2, \langle f_{2,i} \mid i \in I_2 \rangle) \\ M_3 &= (I_3, p_3, r_3, T_3, A_3, B_3, P_3, \langle f_{3,i} \mid i \in I_3 \rangle). \end{aligned}$$

We assume also that the number of cells in the models (given by n for each of the models) is $n_3 \geq 8$ for the CA model M_3, $n_2 = 2n_3$ for the PDE model M_2, and $n_1 = 4n_2$ for the CODE model M_1. Let all the other parameters values set in Section 4 be fixed here.

We define local abstraction tuples

$$\begin{aligned} \gamma_2 &= (\pi_2, \lambda_2, \langle \phi_{2,j}, \psi_{2,j}, \theta_{2,j} \mid j \in I_2 \rangle) \\ \gamma_3 &= (\pi_3, \lambda_3, \langle \phi_{3,j}, \psi_{3,j}, \theta_{3,j} \mid j \in I_3 \rangle) \end{aligned}$$

for comparing the components of the CODE and PDE models M_1 and M_2, and between the components of the PDE and CA models M_2 and M_3 respectively. The local abstraction tuples γ_2 and γ_3 determine abstraction tuples Γ_2 and Γ_3 respectively.

In Section 6.3 we discuss the sense in which the behaviour of M_2 approximates that of M_1 with respect to Γ_2, and how the behaviour of M_2 is approximated by M_3 with respect to Γ_3 by discussion of approximation tuples Ω_2 and Ω_3 respectively. We thus construct the hierarchy illustrated in Figure 13 in the specific case of the CODE, PDE and CA models.

6.1 CODE–PDE component abstraction

The local CODE-PDE component abstraction γ_2 is determined from the PDE model parameter fitting described in Section 4.2.

Spaces. We begin by comparing the sets I_1 and I_2 of cells in the CODE and PDE models M_1 and M_2. We have chosen (see Section 4.2) that each PDE cell represents a strand of 4 CODE cells; we thus define $\pi_2 : I_1 \to I_2$, for all CODE cells $i \in I_1$, by

$$\pi_2(i) = \left\lceil \frac{i}{4} \right\rceil .$$

The inverse $\pi_2^{-1} : I_2 \to \mathit{Powerset}(I_1)$ of π_2 maps single PDE cells to the sets of CODE cells that they represent, and is defined for all $j \in I_2$, by

$$\pi_2^{-1}(j) = \{4j-3, 4j-2, 4j-1, 4j\}.$$

Clocks. In the parameter-fitting of Section 4.2, we stated that each PDE timestep abstracted 5 CODE timesteps. This is formalised by defining a retiming $\lambda_2 : T_1 \to T_2$ for all CODE clock steps $t \in T_1$, by

$$\lambda_2(t) = \left\lfloor \frac{t}{5} \right\rfloor .$$

The retiming λ_2 determines an immersion $\overline{\lambda}_2 : T_2 \to T_1$, defined for all $t \in T_2$ by $\overline{\lambda}_2(t) = 5t$, and the set $Start_{\lambda_2} = \{0, 5, 10, 15, \ldots\}$ comprising those CODE clock cycles that are the first abstracted by each PDE clock cycle.

Observable states. Mapping the voltages between M_1 and M_2 is a simple averaging/scaling operation, determined from the minimum and maximum voltage values that ordinarily occur in each of the models. For the CODE model, a propagating action potential generates minimum and maximum voltage values of -94.25 and 48.25 (mV) respectively; in the PDE model the minimum and maximum values are 0 and 1. The local observable state abstraction maps $\phi_{2,j} : \mathbf{R}^{\pi_2^{-1}(j)} \to \mathbf{R}$ are uniform across I_2. Let $v \in \mathbf{R}^{\pi_2^{-1}(j)}$ be an observable state of the M_1 subspace $\pi_2^{-1}(j) \subseteq I_1$ abstracted by cell $j \in I_2$. We define, $\phi_{2,j}$ for all such v by

$$\phi_{2,j}(v) = \begin{cases} (avg(v) + 94.25)/142.5 & \text{if } avg(v) \in [-94.25, 48.25] \\ 0 & \text{if } avg(v) < -94.25 \\ 1 & \text{if } avg(v) > 48.25 \end{cases}$$

where $avg(v)$ denotes the average value of the observable cell states $v(4j-3), v(4j-2), v(4j-1), v(4j) \in \mathbf{R}$ comprising v. Notice that we have ensured that any value of $avg(v)$ falling outside the interval $[-94.25, 48.25]$ (due to, for example, a large local stimulation or boundary effects) is mapped appropriately to either the minimum or maximum PDE voltage value.

Global hidden states. Each cell in the CODE model has 16 real-valued hidden states representing concentrations of various ionic species and gating variables. In the PDE model, each cell has a single hidden-state which represents an abstract notion usually termed "recovery". It is impossible (and unnecessary) to compare fully all of the (huge number of possible) hidden states of the CODE model with those of the PDE model: many of the complex behaviours of the CODE model are not approximated by the PDE. From Section 5.1 however, we need to ensure that the domain $dom(\psi_{2,j})$ of each local hidden state map $\psi_{2,j} : (\mathbf{R}^{16})^{\pi_1^{-1}(j)} \to \mathbf{R}$ is non-empty. We choose a hidden cell state $rest_c \in \mathbf{R}^{16}$ for which we consider (given a suitable voltage value) a CODE cell to be at rest (such a state is easily found), and let $rest \in (\mathbf{R}^{16})^{\pi_2^{-1}(j)}$ be the uniformly resting state of the M_1 subspace $\pi_2^{-1}(j)$ defined by $rest(4j-3) = rest(4j-2) = rest(4j-1) = rest(4j) = rest_c$. We define $\psi_{2,j}$ to have domain $dom(\psi_{2,j}) = \{rest\}$ comprising only this uniformly resting state. As a rest state is represented in the the PDE model by the value 0, we define $\psi_{2,j}(rest) = 0$.

Parameter streams. The parameter streams in all three models represent only electrical stimuli. Cells in the CODE and PDE models are stimulated by streams from the sets $[T_1 \to \mathbf{R}]$ and $[T_2 \to \mathbf{R}]$ respectively. It is difficult to fully compare streams of stimuli for these two models. Here therefore, we shall simply select two stimulation values from $\mathbf{R}$ for each model: one denoting the absence of stimulation, the other denoting a supra-threshold stimulation for resting tissue. For the CODE model we take the values 0 and -30, representing electrical stimulations of 0nA and 30nA respectively. An appropriate corresponding supra-threshold value for the PDE model is derived in the following way.

First, we compute the minimum duration required for a spatially- and temporally-uniform 30nA stimulation across the four-cell CODE subspace $\pi_2^{-1}(j)$ (for a chosen PDE cell $j \in I_2$) to generate an action potential from a global resting state; the computed value is 1.38ms (i.e. stimulating from timestep 0 to timestep 137). CODE timestep 137 is abstracted by PDE timestep $\lambda_2(137) = 27$. Our required stimulation value is computed as the minimum that initiates an action potential in the PDE model by uniform stimulation of cell j from timestep 0 to timestep 27 (given an initial global resting state); the computed value is 0.12. Note that the choice of j will affect the outcome of this procedure: a shorter/weaker stimulation initiates action potentials in boundary regions. We shall ignore this complication here: the value 0.12 was calculated from cell 250 in a $n_2 = 500$ cell system, and will be used across the space I_2.

We define a local parameter sequence map

$$\theta_{2,j} : (\mathbf{R}^*)^{\pi_2^{-1}(j)} \to \mathbf{R}$$

with domain $dom(\theta_{2,j}) = \{stim, nostim\} \subset (\mathbf{R}^*)^{\pi_2^{-1}(j)}$ where $stim, nostim \in (\mathbf{R}^5)^{\pi_2^{-1}(j)}$ are local stimulation sequences for the subspace $\pi_2^{-1}(j)$, where $stim_t(i) = -30$ and $nostim_t(i) = 0$ for all $1 \leq t \leq 5$ and $i \in \pi_2^{-1}(j)$. This domain restricts the global CODE parameter stream sets that we can compare with those of the PDE model in the following sense: For every PDE cell $j \in I_2$ and for every PDE clock cycle $t \in T_2$, the 20 parameter values supplied to all 4 cells in the subspace $\pi_2^{-1}(j)$ during the 5 clock cycles $\overline{\lambda}(t), \ldots, \overline{\lambda}(t+1) - 1$ must all be equal to 0 or all equal to -30. This restriction still allows us to compare most properties of interest between the two models. We define each $\theta_{2,j}$ simply by

$$\begin{aligned} \theta_{2,j}(stim) &= 0.12 \\ \theta_{2,j}(nostim) &= 0. \end{aligned}$$

6.2 PDE-CA component abstraction

The local PDE-CA abstraction γ_3 is determined from the CA model parameter fitting described in Section 4.3.

Spaces. We first compare the sets I_2 and I_3 of cells in the PDE and CA models M_2 and M_3. We have chosen (see Section 4.3) each CA cell to abstract two neighbouring PDE cells; we thus define $\pi_3 : I_2 \to I_3$, for all PDE cells $i \in I_2$, by

$$\pi_3(i) = \left\lceil \frac{i}{2} \right\rceil .$$

The inverse $\pi_3^{-1} : I_3 \to Powerset(I_2)$ of π_3 maps single CA cells to sets of PDE cells that they abstract, and is defined, for all $j \in I_3$, by

$$\pi_3^{-1}(j) = \{2j - 1, 2j\}.$$

Clocks. From the CA parameter values set in Section 4.3, each CA timestep was assumed to abstract 21.3 PDE timesteps. This gives us a retiming $\lambda_3 : T_2 \to T_3$ defined, for all PDE clock cycles

$t \in T_2$ by

$$\lambda_3(t) = \left\lfloor \frac{t}{21.3} \right\rfloor .$$

The retiming λ determines an immersion $\overline{\lambda}_3 : T_3 \to T_2$ defined for all $t \in T_3$ by $\overline{\lambda}_3(t) = \lceil 21.3t \rceil$ and its range $Start_{\lambda_3} = \{0, 22, 43, 64, 85, \ldots\}$.

Observable states. Mapping the voltages between the PDE and CA models is straightforward. CA cells' observable states 1 and 0 denote an action potential voltage and a non-action potential voltage respectively. In Section 4 we considered a voltage for the CODE model to constitute an action potential if greater than -40mV. Applying the CODE-PDE local abstraction map $\phi_{2,i}$ of the previous section to a local observable CODE state $v \in \mathbf{R}^{\pi_2^{-1}(i)}$ with a uniform value of -40 gives $\phi_{2,i}(v) = 0.38$; we thus take take 0.38 as the delimiting value for the PDE and define local abstraction maps $\phi_{3,j} : \mathbf{R}^{\pi_3^{-1}(j)} \to \{0, 1\}$ for each $j \in I_3$, for all local observable PDE states $v \in \mathbf{R}^{\pi_3^{-1}(j)}$ (comprising cell states $v(2j-1), v(2j) \in \mathbf{R}$) by

$$\phi_{3,j}(v) = \begin{cases} 1 & \text{if } avg(v(2j-1), v(2j)) > 0.38 \\ 0 & \text{if } avg(v(2j-1), v(2j)) \leq 0.38. \end{cases}$$

Hidden states. As with the CODE-PDE abstraction, we avoid the problem of unnecessarily mapping all possible hidden states of PDE cells to the CA model. Let local hidden state abstraction map $\psi_{3,j} : \mathbf{R}^{\pi_3^{-1}(j)} \to \{0, 1, \ldots, B_{max}\}$ for each $j \in I_3$ have domain $dom(\psi_{3,j}) = \{rest\}$ containing only the pair $rest(2j-1) = rest(2j) = 0$ of PDE hidden cell states that compare with the CA's resting state 0; we thus define

$$\psi_{3,l}(rest) = 0.$$

Parameter streams. The CA model's streams supply only two different values: 1 represents a supra-threshold stimulation for resting tissue and 0 represents the absence of stimulation. The corresponding values for the PDE model have been set (see the previous section) at 0.12 and 0. For each $j \in I_3$, we define a local parameter sequence map

$$\theta_{3,j} : (\mathbf{R}^*)^{\pi_3^{-1}(j)} \to \{0, 1\}$$

with the domain $dom(\theta_{3,j}) = \{stim_{21}, stim_{22}, nostim_{21}, nostim_{22}\}(\mathbf{R}^*)^{\pi_3^{-1}(j)} \to \{0, 1\}$ and where $stim_k, nostim_k \in (\mathbf{R}^k)^{\pi_3^{-1}(j)} \to \{0, 1\}$, $stim_{k,i}(t) = 0.12$ and $nostim_k(t) = 0$ for $k = 21, 22$, $i = 2j-1, 2j$ and $t = 1, \ldots, k$ comprising all pairs of sequences of PDE stimuli values that we will compare with CA stimuli values (recall that the retiming λ_3 determines parameter sequences only of lengths 21 and 22 only). Let

$$\begin{array}{rcrcl} \theta_{3,j}(stim_{21}) & = & \theta_{3,j}(stim_{22}) & = & 1 \\ \theta_{3,j}(nostim_{21}) & = & \theta_{3,j}(nostim_{22}) & = & 0. \end{array}$$

6.3 Approximation of observable behaviours

In Section 5.2 we introduced a simple notion of abstraction between two models with respect to component abstraction maps. In Section 6.1 we defined abstraction maps for relating components

of the CODE model with those of the PDE model, and in Section 6.2 we defined maps that relate the PDE and CA models' components. We now consider the sense in which the behaviour of the PDE approximates that of the CODE model, and how the behaviour of the CA approximates that of the PDE.

In Section 4 we illustrated the behaviours of all three models using the simplest of simulations: a single simulation of one end of the model strand to initiate a single propagating action potential. The parameters of the PDE and CA models were fitted to achieve a good match of action potential durations and propagation velocities for all three models, as illustrated in Figures 6, 8 and 9, and to achieve a good match of restitution properties between the CODE and PDE models (Figure 10). It is these parameter values that determined the component abstractions formalised above. The initial states and parameter streams for the PDE simulation were abstracted by Γ_2; these were then abstracted by Γ_3 for the CA simulation. (The initial hidden state and parameter streams chosen for the CODE simulation are members of the domains $dom(\psi_2)$ and $dom(\Theta_2)$ respectively, and their abstraction by Γ_2 fall within the domains $dom(\psi_3)$ and $dom(\Theta_3)$. This ensures that the formal comparison of the models' behaviours can be made.)

To compare formally the three models behaviours, we will use the metric d from Section 5.2, redefined for the sets of global observable states of the two models (M_2 and M_1) where the comparisons will be made: for $k = 2, 3$ we define $d_k : A_k^{I_k} \times A_k^{I_k} \to \mathbf{R}^+$ for comparing states of M_k with abstracted states of M_{k-1} for all $v_1, v_2 \in A_k^{I_k}$, by

$$d_k(v_1, v_2) = \sum_{j \in I_k} \mid v_1(j) - v_2(j) \mid .$$

It can be seen from Figures 6 and 8 that the behaviour (a propagating action potential) of the PDE model approximates that of the CODE with respect to Γ_2 for the single initial state and set of parameter streams given, given reasonable tolerance ε_2. The values of d_2 for this comparison shown in these figures is shown in Figure 14; the plateau at value 28 essentially represents the difference in action potential shape between the two models because the wave-fronts and backs match very closely. It should also be clear from Figure 10 that the PDE model approximates formally the behaviour of the CODE given periodic stimulation for the periods given, for a reasonably small tolerance value ε_2.

Similarly the CA action potential approximates that of the PDE with respect to Γ_3 for a small tolerance value d_3 for the initial state and parameter stream set of Figures 8 and 9. As illustrated in Figure 14, the value of d_3 never rises above 2: this is because the wave-fronts and backs of the two models match under the abstraction Γ_3, and because all local PDE voltage values are mapped exactly onto the two voltage values 0 and 1 of the CA.

These approximations between the three models do not hold for all possible initial states and parameter stream sets. For example, with repetitive stimulation the CA model will not give the changes in action potential duration, and will show different Wenckebach rhythms, and the PDE and CODE models will respond differently to a stimulus applied in the wake of a travelling wave, as the models have different vulnerable windows [24]. However, we can say that the behaviour of the PDE model approximates that of the CODE, and that the behaviour of the CA approximates that

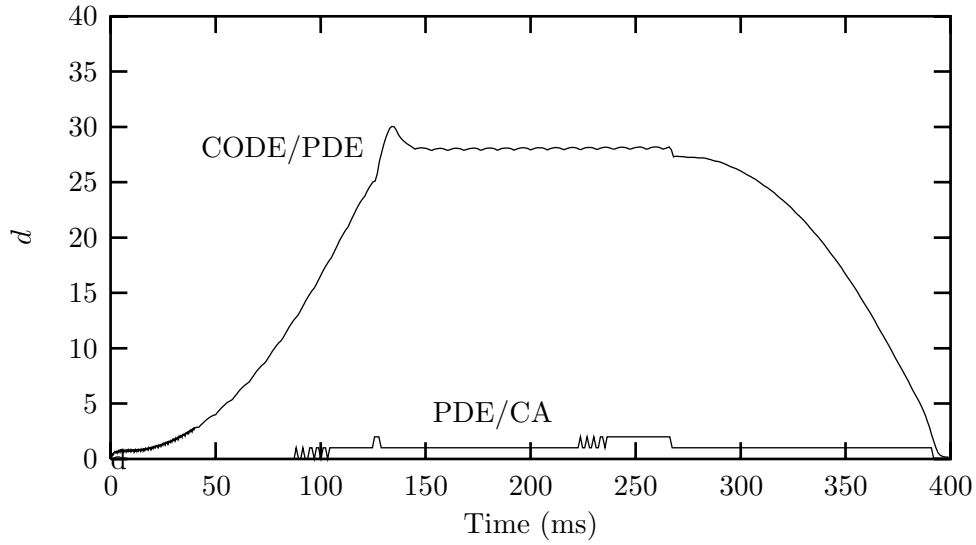

Figure 14: Comparison of the three models with respect to the metrics d_2 and d_3 for the simulations illustrated in Figures 6, 8 and 9.

of the PDE with respect to Γ_2 and Γ_3 for approximation tuples

$$\begin{aligned}\Omega_2 &= (d_2, \varepsilon_2, sub(T_1), sub([T_1 \to P_1]^{I_1}), sub(A_1^{I_1}), sub(B_1^{I_1}))\\ \Omega_3 &= (d_3, \varepsilon_3, sub(T_2), sub([T_2 \to P_2]^{I_2}), sub(A_2^{I_2}), sub(B_2^{I_2})).\end{aligned}$$

for non-trivial choices of subsets and for small values of ε_2 and ε_3.

7 Multi-level models of cardiac tissue

In Section 5, we saw how different mathematical models, operating on different time and space scales, could be compared formally, and we defined formally a notion that one model's behaviour is an abstraction of that of a another model.

In this section we show how different models can be combined into *hierarchical* or *multi-level systems*, that preserve some essential behaviours of cardiac tissue, as well as being computationally practical. Specifically, we define a method for building multi-level models from a collection of component models. Our technique results in hybrid models comprised of cells of different component models that compute on different data sets with respect to different clocks. Neighbouring cells of different types communicate with respect to the abstraction maps of Section 5 that relate them.

In Section 7.1 we collect together the component models and abstraction maps required to construct a multi-level model. In Section 7.2 we show how the multi-level model is constructed using either a bottom-up or top-down technique. In Section 7.3 we define formally the operation of a multi-level model, using equations that are generalisations of the SCA equations of Section 3.2, to

allow for the multiple clocks and data sets.

7.1 Component hierarchy

Consider $m \geq 2$ models $M_1, \ldots, M_m$ of cardiac tissue, where each model M_k is specified by

$$M_k = (I_k, p_k, r_k, T_k, A_k, B_k, P_k, \langle f_{k,i} \mid i \in I_k \rangle).$$

for $1 \leq k \leq m$. Let the components of model M_{k-1} be related to those of model M_k (for $2 \leq k \leq m$) with respect to abstraction tuple Γ_k which we express in its local form:

$$\gamma_k = (\pi_k, \lambda_k, \langle \phi_{k,j}, \psi_{k,j}, \theta_{k,j} \mid j \in I_k \rangle).$$

For $2 \leq k \leq m$, let the observable behaviour of model M_k approximate the observable behaviour of model M_{k-1} with respect to abstraction tuple Γ_k for some formal notion of approximation Ω_k.

Of the abstraction maps, the space abstraction maps π_k (and their inverses π_k^{-1}) are used in the *construction* of a multi-level model. The retimings λ_k (and immersions $\overline{\lambda}_k$) and the local observable state abstraction maps $\phi_{k,j}$ are used to specify the *operation* of the model. Specifically, the immersions and local observable state abstraction maps determine, for a cell's local computation, the timing and state conversions required to read observable states from neighbourhoods comprising more-detailed cells. The retimings determine the timing for reading from neighbours that are more-abstract than the cell itself. To convert a neighbour's observable state in this case, we need to define a *reverse* map for $\phi_{k,j}$ for $2 \leq k \leq m$ and $j \in I_k$. These take the form

$$\overline{\phi}_{k,j} : A_k \to A_{k-1}$$

and must be defined such that, for all observable states $v \in A_k$ of cell $j \in I_k$,

$$\phi_{k,j}((\overline{\phi}_{k,j}(v) \mid i \in \pi_k^{-1}(j)) = v$$

such that abstracting the uniform state of the M_{k-1} subspace $\pi_k^{-1}(j) \subseteq I_{k-1}$, whose cells values are all reverse abstractions of M_k cell state $v \in A_2$, gives the value v itself.

7.2 Multi-level model construction

Constructing a multi-level model M from the component models $M_1, \ldots, M_m$, the abstraction tuples $\Gamma_2, \ldots, \Gamma_m$ and reverse maps $\phi_{k,j}$ for $2 \leq k \leq m$ and $j \in I_k$, is achieved by operating on the component space sets $I_1, \ldots, I_m$ using the space abstraction maps $\pi_2, \ldots, \pi_m$ and their inverses. First, let J denote the disjoint union of all the component space sets,

$$J = \bigcup_{1 \leq k \leq m} \{(k, i) \mid i \in I_k\}$$

which comprises all cells that *can* exist in model M. A cell $(k, i) \in J$ existing in the multi-level model M is a *copy* of the M_k cell $i \in I_k$ in that it computes with resect to clock T_k on data from

the sets A_k and B_k, has local dynamics $f_{k,i}$, and takes input parameter values from a stream of type $[T_k \to P_k]$. Its neighbourhood is also, in some sense (see below), equivalent to that of the cell $i \in I_k$.

A space set $I \subset J$ of all cells *actually existing* in M includes exactly one representative, with respect to the $\pi_2, \ldots, \pi_m$, of each cell $i \in I_1$ of the most detailed model M_1. Formally, I is a valid space set for M if for all $i \in I_1$,

$$(1, i) \in I \quad \text{xor} \quad (2, \pi_2(i)) \in I \quad \text{xor} \quad (3, \pi_3(\pi_2(i))) \in I \quad \text{xor} \quad \ldots \quad \text{xor} \quad (m, \pi_m(\cdots(\pi_2(i)\cdots)) \in I.$$

Two natural approaches to multi-level model construction are *bottom-up* and *top-down* replacement of cells.

Bottom-up construction. We begin with a copy of the most detailed model M_1 by setting $I = \{1\} \times I_1$ (note that this is a valid space set with respect to the above condition). We repeat the operation of replacing any existing sub-space $\{k-1\} \times \pi_k^{-1}(j) \subseteq I$ of M (comprising only M_{k-1} cells) with the single cell (k, j) that abstracts it in model M_k. This operation, illustrated in the one-dimensional case in Figure 15 gives a new valid space set

$$I' = (I - (\{k-1\} \times \pi_k^{-1}(j))) \cup \{(k, j)\}$$

from which we can preform further replacements.

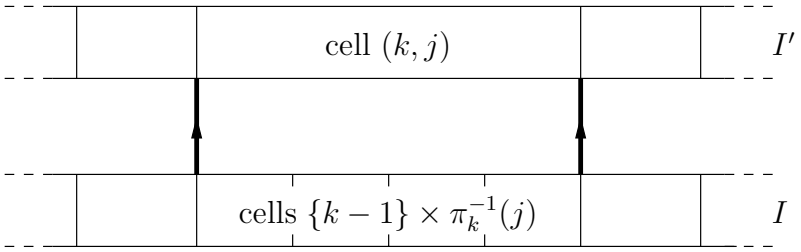

Figure 15: A bottom-up operation in a one-dimensional multi-level model.

Top-down construction. This is reverse of bottom-up constriction. We begin with a copy of the most abstract model M_m by setting $I = \{m\} \times I_m$ (again notice that this is valid with respect to the formal condition above). We then repeat the operation of replacing any single model M_k cell $(k, j) \in I$ of M with the cells $\{k-1\} \times \pi_k^{-1}(j) \subseteq I$ (of model M_{k-1}) to give a new valid set

$$I' = (I - \{(k, j)\}) \cup (\{k-1\} \times \pi_k^{-1}(k, j))$$

as illustrated in the one-dimensional case by Figure 16.

7.3 Multi-level model operation

We now define formally the operation of a multi-level model M constructed from the components and abstraction maps above, and with space set I. The operation of M is determined by the data and local dynamics of the component models, and the space, time, and local observable state

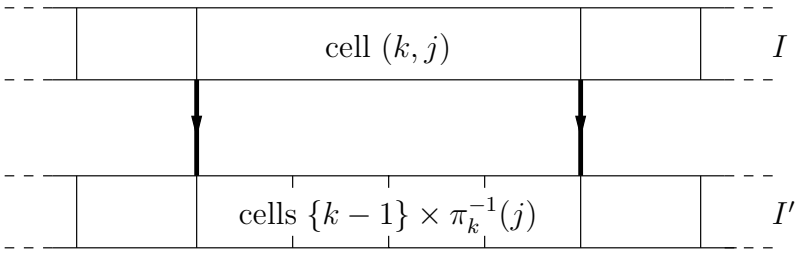

Figure 16: A top-down operation in a one-dimensional multi-level model.

abstraction and reverse maps. We first formalise the sets of global states and parameter streams, and the architecture of the system.

Tissue state. Let

$$Obs(M) \subseteq \left(\bigcup_{1 \leq k \leq m} A_k \right)^I$$

denote the set of all valid global observable states of M where the cell state $v(k, i)$ of each cell $(k, i) \in I$ is in the set A_k of possible M_k cell states. Formally, $v \in Obs(M)$ if, and only if, $v(k, i) \in A_k$ for all $(k, i) \in I$. Similarly, let

$$Hid(M) \subseteq \left(\bigcup_{1 \leq k \leq m} B_k \right)^I$$

denote the set of all valid global hidden states of M, so that $u \in Hid(M)$ if, and only if, $u(k, i) \in B_k$ for all $(k, i) \in I$.

Parameter streams. Let $Param(M)$ denote the set of all valid input parameter streams to M. That is, $Param(M)$ is the set of all stream-sets of the form

$$a = (a_{k,i} \in [T_k \to P_k] \mid (k, i) \in I)$$

such that each model M_k cell $(k, i) \in I$ has associated with it a stream $a_{k,i}$ of data from P_k clocked by T_k.

Architecture. Given an observable state $v \in Obs(M)$, we can determine *approximate* observable states for all those cells $(k, i) \in J - I$ that exist in the component models but *not* in the multi-level model M, using the local subspace abstraction mappings $\phi_{k,j}$ and the reverse mappings $\overline{\phi}_{k,j}$. This fact is central to the operation of M and leads to the following simple definition of its architecture. Let each cell $(k, i) \in I$ have neighbourhood $nhd(k, i) = \{r(k, i, 1), r(k, i, 2), \ldots, r(k, i, p(i))\} \subseteq J$ (some members of which may not be in I) enumerated by architecture functions $p : I \to \mathbf{N}$ and $r : I \times \mathbf{N} \to J$ defined directly from the component models, for all cells $(k, i) \in I$ by $p(k, i) = p_k(i)$, and for $1 \leq j \leq p_k(i)$, by $r(k, i) = (k, r_k(i))$.

Model equations. Each cell $(k, i) \in I$ of M has associated with it a local state function

$$F_{k,i} : T_k \times Param(M) \times Obs(M) \times Hid(M) \to S_k$$

where $F_{k,i}(t,a,v,u) \in S_k$ is the state of cell (k,i) at time $t \in T_k$ given parameter stream sets $a \in Param(M)$ and initial states $v \in Obs(M)$ and $u \in Hid(M)$, and where $S_k = A_k \times B_k$. We denote the observable part of $F_{k,i}(t,a,v,u)$ by $V_{k,i}(t,a,v,u) \in A_k$ in the same way as for the basic SCA model of Section 3.2. To simplify the definition of the operation of M we also associate with each *non-existing* cell $(k,i) \in J - I$ an *approximated observable state function*

$$V_{k,i} : T_k \times Param(M) \times Obs(M) \times Hid(M) \to A_k$$

where $V_{k,i}(t,a,v,u)$ is the approximated state of (k,i) at time t given parameters a and initial state (v,u).

We first define each $F_{k,i}$ for $(k,i) \in I$ by

$$\begin{aligned} F_{k,i}(0,a,v,u) &= (v(k,i), u(k,i)) \\ F_{k,i}(t+1,a,v,u) &= f_{k,i}(F_{k,i}(t,a,v,u), V_{r(k,i,1)}(t,a,v,u), \ldots, V_{r(k,i,p(k,i))}(t,a,v,u), a_{k,i}(t)). \end{aligned}$$

Notice the similarity between this equation and those for the general single-level model of Section 3.2. Note also that each of the $V_{r(k,i,l)}$ functions (for $l = 1, \ldots, p(k,i,l)$) on the right-hand side of the equation may either be observable coordinates of the local state function $F_{r(k,i,l)}$ in the case that the cell $r(k,i,l) \in I$ exists in the model, or approximated observable state functions in the case that the cell $r(k,i,l) \in J - I$ does not exist in the model.

The approximated observable states $V_{k,i}(t,a,v,u)$ for each non-existing cell $(k,i) \in J - I$ are calculated recursively as follows:

(i) if (k,i) abstracts (via π_k^{-1}, π_{k-1}^{-1}, etc) more-detailed cells in I, then we abstract using $\phi_{k,i}$ the observable state of the M_{k-1} subspace $\{k-1\} \times \pi_k^{-1}(i) \subset J$ at clock cycle $\overline{\lambda}_k(t)$ that t abstracts; otherwise

(ii) (k,i) is abstracted (via π_{k+1}, π_{k+2}, etc) by a cell in I, and we apply the reverse map $\overline{\phi}_{k+1,\pi_{k+1}(i)}$ to the observable state of M_{k+1} cell $(k+1, \pi_{k+1}(i)) \in J$ at clock cycle $\lambda_{k+1}(t)$ that abstracts t.

Formally, we define each $V_{k,i}$ for $(k,i) \in J - I$, by

$$V_{k,i}(t,a,v,u) = \begin{cases} \phi_{k,i}(V_{k-1,j}(\overline{\lambda}_k(t), a, v, u) \mid j \in \pi_k^{-1}(i)) & \text{if } (k,i) \in L \\ \overline{\phi}_{k+1,\pi_{k+1}(i)}(V_{k+1,\pi_{k+1}(i)}(\lambda_{k+1}(t), a, v, u)) & \text{otherwise} \end{cases}$$

where $L \subseteq J - I$ is the set of all non-existing cells that abstract more-detailed cells that are in I, and is defined as follows: all cells $(m,i) \in J - I$ are in L; and for $k < m$, if $\pi_{k+1}(i) \in L$ and $(k,i) \notin I$, then (k,i) is in L; no other cells are in L.

Consider, for example, the $m = 3$-level one-dimensional model M depicted in Figure 17, where the space abstractions $\pi_k : I_{k-1} \to I_k$ (for $k = 2, 3$) are defined by $\pi_k(i) = \lceil i/2 \rceil$ for all $i \in I_{k-1}$. The set J of all possible cells comprises all cell indices in the figure. The sets I (outlined in bold) and L are given by

$$\begin{aligned} I &= \{((1,1),(1,2),(1,9),(1,10),(2,2),(2,6),(3,2)\} \\ L &= \{(2,1),(2,5),(3,1),(3,3)\}. \end{aligned}$$

The Figure shows all observable state abstractions, reverse-abstractions and their timings required for the existing cells to compute, assuming a nearest neighbour architecture. For example, the computation at cell $(3,2)$ at time $t \in T_3$ requires observable voltage values from its neighbours $(3,1)$ and $(3,3)$. Neither of these cells exist in M, and therefore their values must be approximated. Consider the voltage for cell $(3,1)$ at time t. This is approximated by $\phi_{3,1}$ from the local observable state of the subspace $\{(2,1),(2,2)\}$ of time $\overline{\lambda}_{3,3}(t) \in T_2$. However, cell $(2,1)$ does not exist in M and therefore its observable state at time $\overline{\lambda}_{3,3}(t)$ also needs to be approximated ; this is achieved by abstracting the observable state of the subspace $\{(1,1),(1,2)\}$ of time $\overline{\lambda}_{2,1}(\overline{\lambda}_{2,1}(t)) \in T_1$ using $\phi_{2,1}$.

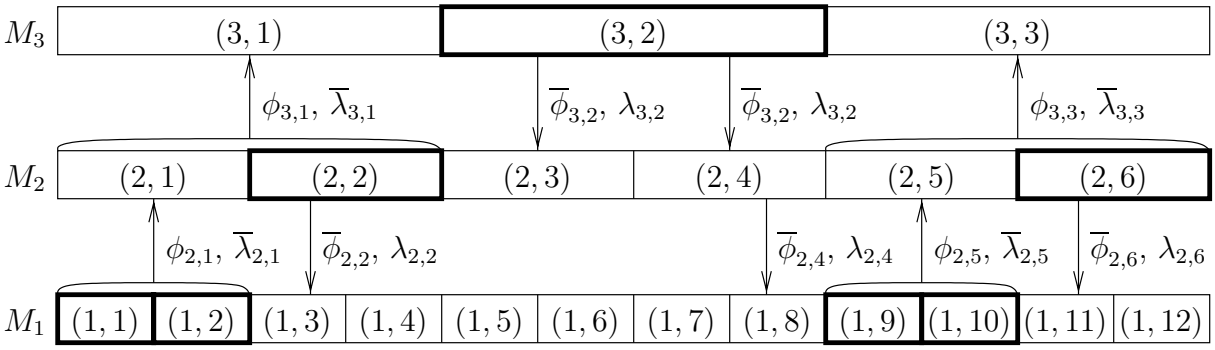

Figure 17: Operation of a 3-level model.

8 Examples of multi-level models

We now apply our general construction technique to build a multi-level model of electrical activity along a one-dimensional strand of cardiac tissue. We use the three models defined in Section 4 as components, using the component abstraction mappings of Section 6 that compare them. Let M_1 be the n_1-cell CODE Oxsoft model of Section 4.1, let M_2 be the n_2-cell PDE model of Section 4.2 and let M_3 be the n_3-cell CA model of Section 4.3.

We shall use the bottom-up method of construction and thus our initial model is a copy of the Oxsoft CODE model M_1 and the initial set of cells is $I = \{1\} \times I_1$. We begin, in Section 8.1, by replacing all but a central region of the Oxsoft CODE model M_1 cells by cells from Aliev and Panfilov's PDE model M_2. We are thus adopting a foveal approach to simulation where we have a detailed, fine-grained description of some central region, and more abstract coarse-grained descriptions of activity in the surrounding areas. We investigate the behaviour of this two–level model using the single action potential simulation of Section 4. In Section 8.2, we replace most of the PDE cells in the two level-model by cells from the CA model M_3 and investigate the behaviour of this three-level system using a similar simulation.

8.1 A CODE—PDE two-level model

We first abstract from the base CODE model by replacing all but a few of its cells by PDE cells. The aim of the resulting model is to reconstruct the global spatio-temporal behaviour of the strand at the level of the PDE model, and the temporal behaviour of a small region at the more detailed CODE level. In order to give accurate temporal behaviour for at least one CODE cell, we require that it is surrounded by a locale of other CODE cells in this two-level model; this will "damp-out" any problems due to the coarse time and space granularity (with respect to that of the CODE cells) of adjoining PDE cells.

Recall from Section 7.1 that we require a local reverse abstraction $\overline{\phi}_{2,j} : \mathbf{R} \to \mathbf{R}$ for all PDE cells $j \in I_2$ for mapping PDE cell voltages to CODE cells. We define $\overline{\phi}_{2,j}$ for all PDE voltage values $v \in \mathbf{R}$ by

$$\overline{\phi}_{2,j}(v) = \begin{cases} 142.5v - 94.25 & \text{if } v \in [0,1] \\ -94.25 & \text{if } v < 0 \\ 48.25 & \text{if } v > 1 \end{cases}$$

which we note conforms to the reverse-abstraction requirement of Section 5.1.

For the two-level model to comprise some number $1 \leq q < n_2/2$ of PDE cells at both the left and right ends, leaving a central region of $n_1 - 8q$ CODE cells, we replace each CODE subspace

$$\{1\} \times \pi_2^{-1}(j) = \{(1, 4j-3), (1, 4j-2), (1, 4j-1), (1, 4j)\}$$

by the single PDE cell $(2, j)$, for $j = 1, \ldots, q$ and for $j = n_2 - q + 1, \ldots, n_2$. This bottom-up operation is illustrated in Figure 18 and gives space set

$$\begin{aligned} I \;=\; & \{(2,1), (2,2), \ldots, (2,q), \\ & (1, 4q+1), (1, 4q+2), \ldots, (1, n_1 - 4q), \\ & (2, n_2 - q + 1), (2, n_2 - q + 1), \ldots, (2, n_2)\}. \end{aligned}$$

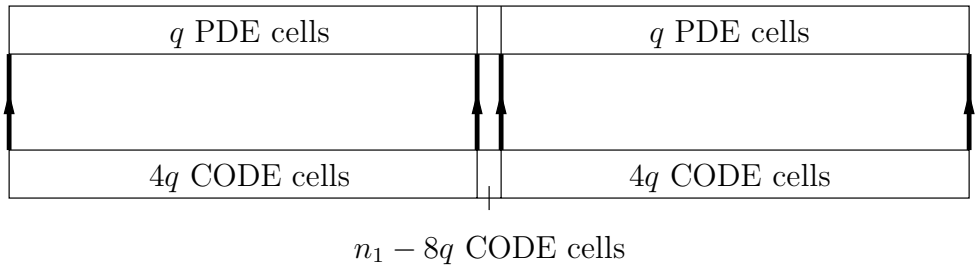

Figure 18: Bottom-up construction of the CODE–PDE two-level model.

Model equations. Each cell $(k, i) \in I$ in M has local state function

$$F_{k,i} : T_k \times Param(M) \times Obs(M) \times Hid(M) \to S_k$$

determined from two models M_1 and M_2, the abstraction tuple Γ_2, the reverse maps $\overline{\phi}_{2,j}$ for $j \in I_2$ and the space set I, as follows where we have expanded the definitions of the approximated observable state functions.

At time 0. For all cells $(k, i) \in I$,

$$F_{k,i}(0, a, v, u) = (v(k, i), u(k, i)).$$

For the CODE cells, at time $t + 1 \in T_1$. For cell $(1, 4q + 1)$ adjoining the left PDE region,

$$\begin{aligned} F_{1,4q+1}(t + 1, a, v, u) \;=\; & f_{1,4q+1}(F_{1,4q+1}(t, a, v, u), \\ & \overline{\phi}_{2,q}(V_{2,q}(\lambda_2(t), a, v, u)), \\ & V_{1,4q+2}(t, a, v, u), a_{1,4q+1}(t)). \end{aligned}$$

(a similar equation applies for cell $(1, n_1 - 4q)$). For all other CODE cells $(1, i)$,

$$F_{1,i}(t + 1, a, v, u) = f_{1,i}(F_{1,i}(t, a, v, u), V_{1,i-1}(t, a, v, u), V_{1,i+1}(t, a, v, u), a_{1,i}(t)).$$

For the PDE cells, at time $t + 1 \in T_2$. For the left-most cell $(2, 1)$,

$$F_{2,1}(t + 1, a, v, u) = f_{2,1}(F_{2,1}(t, a, v, u), V_{2,2}(t, a, v, u), a_{2,1}(t))$$

(a similar equation applies for the right-most cell $(2, n_2)$). For cell $(2, q)$ adjoining the CODE region,

$$\begin{aligned} F_{2,q}(t + 1, a, v, u) \;=\; & f_{2,q}(F_{2,q}(t, a, v, u), V_{2,q-1}(t, a, v, u), \\ & \phi_{2,q+1}(V_{1,i}(\overline{\lambda}_2(t), a, v, u) \mid i \in \{4q + 1, 4q + 2, 4q + 3, 4q + 4\}), \\ & a_{2,q}(t)) \end{aligned}$$

(similarly for cell $(2, n_2 - q + 1)$). For all other PDE cells $(2, i)$,

$$F_{2,i}(t + 1, a, v, u) = f_{2,i}(F_{2,i}(t, a, v, u), V_{2,i-1}(t, a, v, u), V_{2,i+1}(t, a, v, u), a_{2,i}(t)).$$

Hierarchical model behaviour. To illustrate the behaviour of the model, consider a 160mm strand of tissue. In Section 4 we saw that this was modelling by a CODE system of $n_1 = 2000$ cells and a PDE system of $n_2 = 500$ cells. Let us consider a two-level model comprising left and right regions of $q = 245$ PDE cells, and a central region of $n_1 - 8q = 40$ CODE cells. Figure 19 illustrates a simulation involving the stimulation of the leftmost two PDE cells (as in Figure 8) to obtain a single propagating action potential. Notice that this is almost identical to Figure 8 in that wavefront locations and action potential widths and durations match at each snapshot; the global behaviour of the PDE model is thus preserved by the two-level model. Figure 20 shows the voltage, i_{Ca}, i_{Na} and i_K currents of the central CODE cell (1,1000); this is very similar to the behaviour exhibited by the CODE model of Section 4.1 illustrated in Figure 7, and thus the biologically detailed temporal behaviour (of the central CODE cell) is also preserved in the two-level model.

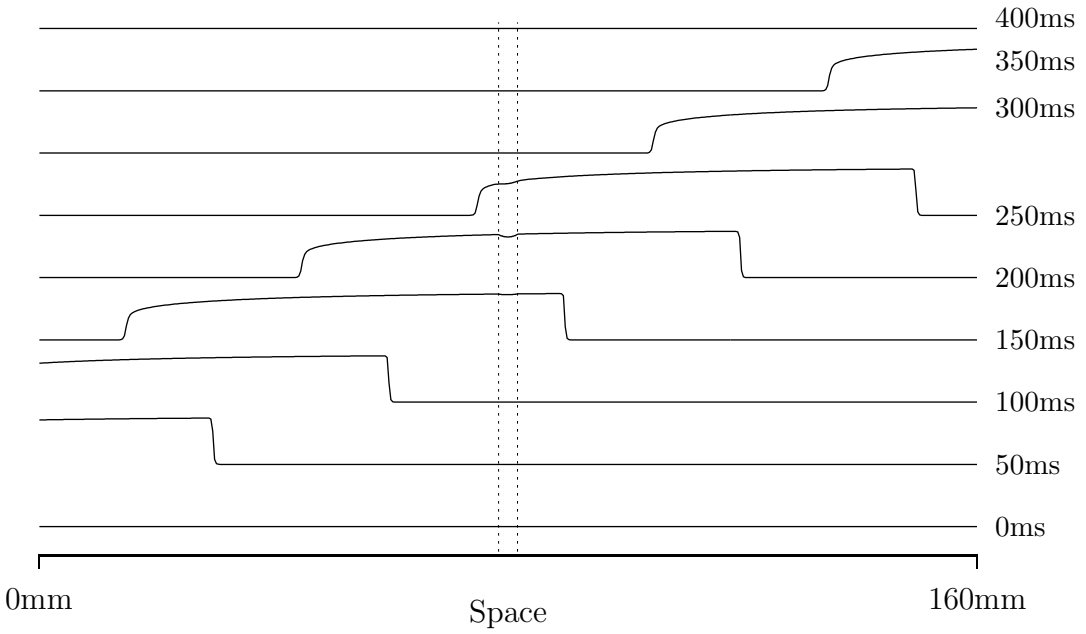

Figure 19: An action potential travelling from left-to-right along The CODE–PDE two-level model following a single stimulus at the left-most PDE cell. The dotted lines denote the boundaries of the central CODE region.

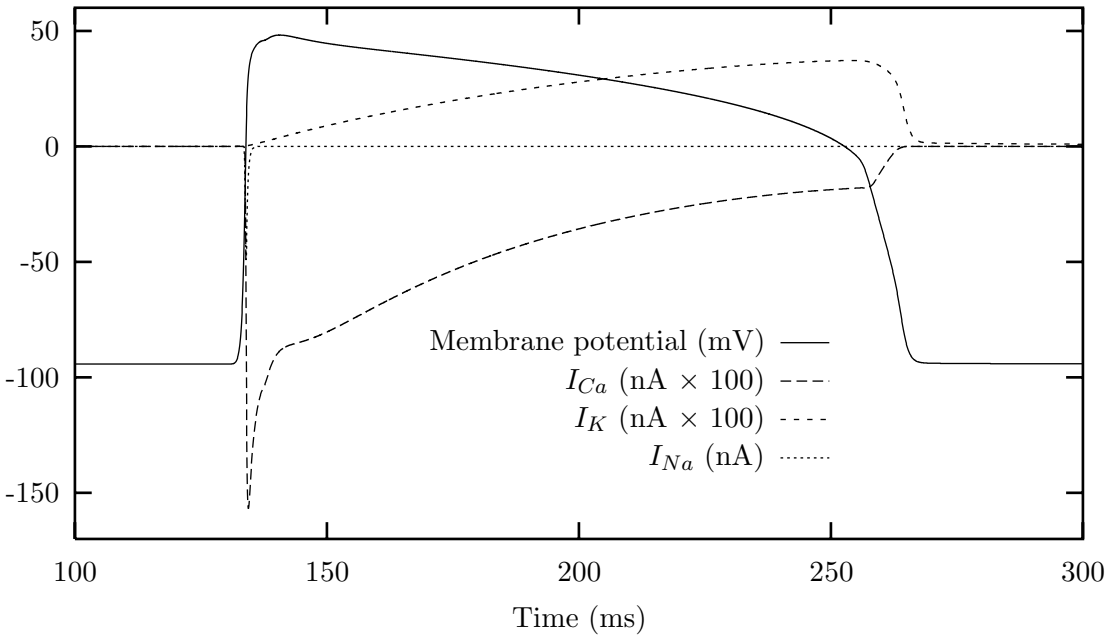

Figure 20: The membrane potential, i_{Ca}, i_{Ca} and i_K currents of the central ODE cell during the middle portion of the simulation of Figure 19.

8.2 A CODE—PDE—CA three-level model

We now abstract from the CODE—PDE two-level model by replacing most of the PDE region by CA cells. The aim of the resulting model is to reconstruct the global spatio-temporal behaviour of the strand at the level of the CODE model, and the temporal behaviour of a small region at the more detailed CODE level. If we replaced all the PDE cells by CA cells the (extreme) coarseness of the CA model might adversely affect the behaviour of the 16 CODE cells in the centre of the model. We will therefore leave a sequence of PDE cells on either side of the CODE region in order to smooth out the effects of the CA cells on the CODE cells.

We first define valid observable state reverse mappings $\overline{\phi}_{3,j} : \{0,1\} \to \mathbf{R}$ for each CA cell $(3,j) \in I_3$ for all CA voltage values $v \in \{0,1\}$ by

$$\overline{\phi}_{3,j}(v) = v.$$

For the three-level model M to comprise $1 \leq s < q/2$ CA cells at both the left and right ends, leaving a central region of $n_1 - 8q$ CODE cells surrounded by $q - 2s$ PDE cells on either side, we replace each PDE subspace

$$\{2\} \times \pi_2^{-1}(j) = \{(2, 2j-1), (2, 2j)\}$$

by the single CA cell $(3,j)$, for $j = 1, \ldots, s$ and for $j = n_3 - s + 1, \ldots, n_3$. This bottom-up operation is illustrated in Figure 21 and gives space set I

$$\begin{aligned} I \;=\; & \{(3,1),(3,2),\ldots,(3,s), \\ & (2,2s+1),(2,2s+2),\ldots,(2,q) \\ & (1,4q+1),(1,4q+2),\ldots,(1,n_1-4q), \\ & (2,n_2-q+1),(2,n_2-q+1),\ldots,(2,n_2-2s), \\ & (3,n_3-s+1),(3,n_2-s),\ldots,(3,n_3)\}. \end{aligned}$$

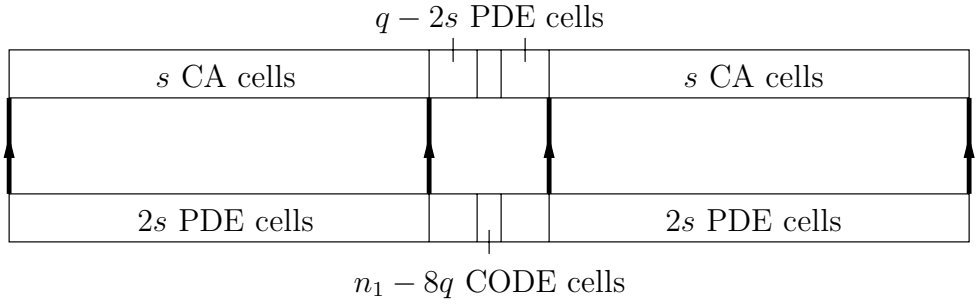

Figure 21: Bottom-up construction of the CODE–PDE-CA three-level model.

Model equations. Each cell $(k,i) \in I$ has local state function

$$F_{k,i} : T_i \times Param(M) \times Obs(M) \times Hid(M) \to S_k$$

determined from models M_1, M_2 and M_3, abstraction tuples Γ_2 and Γ_3, the reverse maps $\overline{\phi}_{2,j}$ for $j \in I_2$ and $\overline{\phi}_{3,j}$ for $j \in I_3$ and the space set I, as follows, where we have expanded the definitions of the approximated observable state functions.

At time 0. For all cells $(k, i) \in I$,

$$F_{k,i}(0, a, v, u) = (v(k, i), u(k, i))$$

For the CODE cells, at time $t + 1 \in T_1$. The equations here are identical to those of the two-level model in Section 8.1.

For the PDE cells, at time $t+1 \in T_2$. For the left-most cell $(2, 2s+1)$ adjoining the left CA region,

$$\begin{aligned} F_{2,2s+1}(t+1, a, v, u) &= f_{2,2s+1}(F_{2,2s+1}(t, a, v, u), \\ &\qquad \overline{\phi}_{3,s}(V_{3,s}(\lambda_3(t), a, v, u)), \\ &\qquad V_{2,2s+2}(t, a, v, u), a_{2,2s+1}(t)) \end{aligned}$$

(a similar equation applies for the right-most PDE cell $(2, n_2 - 2s)$). The remaining PDE cells, including those adjoining the CODE region, have equations identical to those in Section 8.1.

For the CA cells, at time $t + 1 \in T_3$. For the left-most cell $(3, 1)$,

$$F_{3,1}(t+1, a, v, u) = f_{3,1}(F_{3,1}(t, a, v, u), V_{3,2}(t, a, v, u), a_{3,2}(t))$$

(a similar equation applies for right-most cell $(3, n_3)$. For the cell $(3, s)$ adjoining the left PDE region,

$$\begin{aligned} F_{3,s}(t+1, a, v, u) &= f_{3,s}(F_{3,s}(t, a, v, u), V_{3,s-1}(t, a, v, u), \\ &\qquad \phi_{3,s+1}(V_{2,i}(\overline{\lambda}_3(t), a, v, u) \mid i \in \{2s+1, 2s+2\}), \\ &\qquad a_{3,s}(t)) \end{aligned}$$

(a similar equation applies for cell $(3, n_3 - s + 1)$. For all other CA cells $(3, i)$,

$$F_{3,i}(t+1, a, v, u) = f_{3,i}(F_{3,i}(t, a, v, u), V_{3,i-1}(t, a, v, u), V_{3,i+1}(t, a, v, u), a_{3,i}(t)).$$

Hierarchical model behaviour. We illustrate the three-level model's behaviour in the case of a 160 mm length of tissue, represented in the three component models using $n_1 = 2000$, $n_2 = 500$ and $n_3 = 250$ cells. Consider a three-level model comprising left and right regions of $s = 100$ CA cells, intermediate left and right regions of $q - 2s = 45$ PDE cells, and a central region of $n_1 - 8q = 40$ CODE cells. Figure 22 illustrates our action potential simulation produced by stimulating the leftmost CA cell as in Figure 9. Notice how the intermediate PDE regions smooth-out the coarse action potential of the CA. Notice that this is almost identical to Figure 9; the global behaviour of the CA model is preserved by the three-level model. Figure 23 shows the voltage, i_{Ca}, i_{Na} and i_K currents of the central CODE cell (1,1000); this is very similar to the behaviour exhibited by the CODE model of Section 4.1 illustrated in Figure 7, and thus the central CODE cell's detailed behaviour is also preserved by the three-level model.

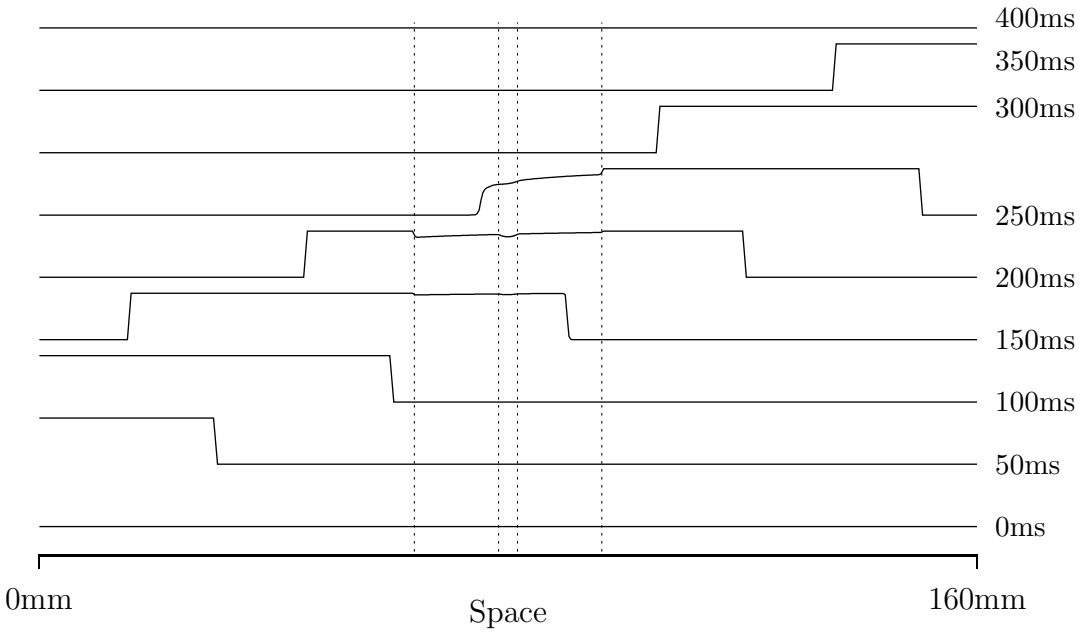

Figure 22: An action potential travelling from left-to-right along The CODE–PDE-CA three-level model following a single stimulus at the left-most CA cell. The dotted lines denote the boundaries of the central CODE region and the intermediate PDE regions.

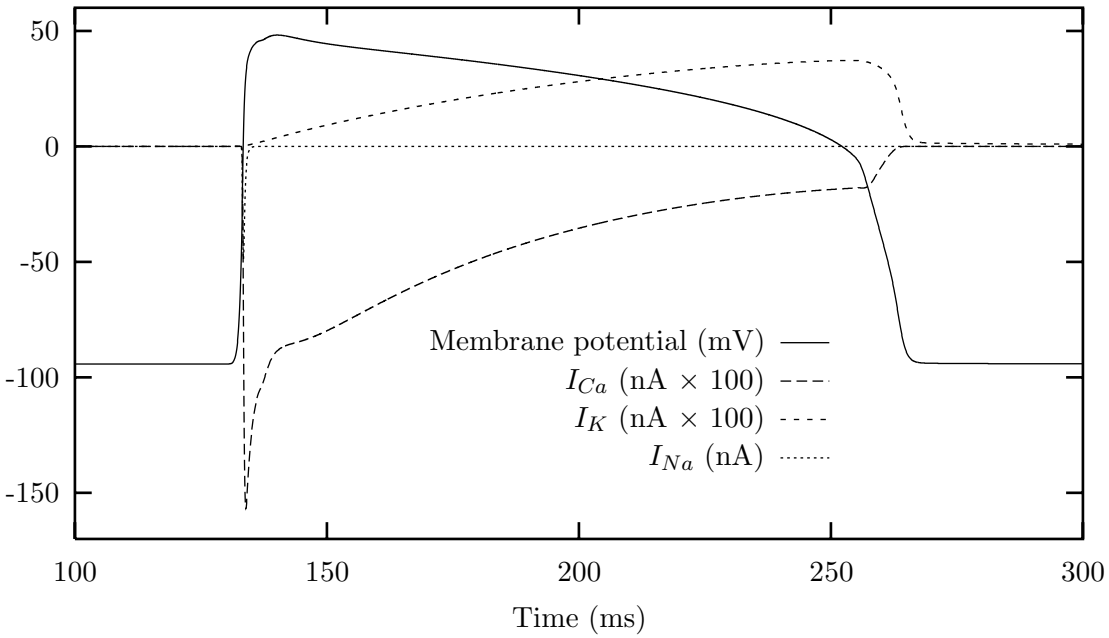

Figure 23: The membrane potential, i_{Ca}, i_{Ca} and i_K currents of the central ODE cell during the middle portion of the simulation of Figure 22.

9 Conclusions

Let us summarise contributions in this paper and point out further directions for research.

Problem of Hierarchy and SCAs. To develop a hierarchical quantitative understanding of the heart it is necessary to use different types of mathematical model and to be able to compare and integrate them in multi-level models that are mathematically hybrid. A unifying mathematical framework is needed which encompasses both the original models and the new multi-level models. We have proposed that the basis for this framework is the class of algorithms that are used for simulation; namely, synchronous concurrent algorithms.

Multi-level SCA models of cardiac tissue. We have analysed the hierarchical structure of SCAs and presented concepts (such as space and time abstraction maps), and made precise the means of comparing different mathematical models at different levels of abstraction that allows us to define the concept of a *multi-level SCA*. We have illustrated these ideas using three models of one-dimensional strands of cardiac tissue (a CODE, a PDE and a CA model) and two multilevel models constructed from them.

The models were designed to capture certain properties of waves a different levels of mathematical abstraction, from high-level "caricature" of the CA model to the detailed, biophysically derived CODE model. The properties reconstructed by these models are summarised in the following table.

Local Properties	**CODE**	**PDE**	**CA**	**CODE/PDE central/elsewhere**	**CODE/PDE/CA central/elsewhere**
action potential shape	accurate	approx	poor	accurate/approx	accurate/poor
restitution	accurate	approx	-	accurate/approx	accurate/-
biophysics	approx	-	-	approx/-	approx/-
Global Properties					
wave propagation	accurate	approx	poor	approx	poor
vulnerability	accurate	poor	poor	poor	poor
dispersion	accurate	poor	-	poor	-
Execution time	4h 53m 51s	11.71s	0.07s	5m 54s	5m 43s

Thus, from the table, we see that our multilevel models (i) preserve the local behaviour of the most detailed component model within the region of interest; (ii) preserve the global behaviour of the most abstract component model throughout the space; and (iii) significantly reduces computation times. (The simulations were performed on a SUN SPARCstation 5 running Solaris 2.5; the algorithms were coded in the C programming language).

Further research. The computational biology of the whole-heart at a biophysically detailed level is not possible and nor would it be necessarily biologically informative. We have shown that the approach using multi-level SCA models is a feasible method for "locally accurate" and large-scale computational modelling. By building on the foundation of the general theory and the demonstration in one-dimension here, we propose to construct a family of whole-heart multilevel SCA models. First, experiments with multilevel SCA models of the ventricles based on the Hunter spatial model [22] will be constructed (we will generalise the single-level models studied in [14]). Secondly, we propose

to extend the general theory to allow the spatial region of interest in a multilevel SCA to become mobile. The aim is to provide systematic methods for tracking meandering waves in multilevel whole-heart SCA models. Specifically, a detailed model at the wave break is combined with an intermediate model on the wavefront and a simple model far from the wavefront.

The approach and theoretical tools are well suited to problems in computational cardiology. However, the methods can be applied to many other excitable media and other systems, biological and otherwise.

Acknowledgements

This work is supported by grant GR/L71544 from UK EPSRC Applied Nonlinear Mathematics Committee, and ESPRIT WG NADA 8533.

References

[1] R R Aliev and A V Panfilov. A simple two-variable model of cardiac excitation. *Chaos, Solitons and Fractals*, 7:293–301, 1996.

[2] V N Biktashev and A V Holden. Control of re-entrant activity in a model of mammalian atrial tissue. *Proceedings of the Royal Society (London) B*, 260:211–217, 1995.

[3] V N Biktashev and A V Holden. Re-entrant activity and its control in a model of mammalian ventricular tissue. *Proceedings of the Royal Society (London) B*, 263:1373–1382, 1996.

[4] V N Biktashev and A V Holden. Re-entrant waves and their control in a model of mammalian ventricular tissue. *Chaos*, 8:48–56, 1998.

[5] J Blom, A V Holden, M J Poole, J V Tucker, and H Zhang. Caress II: a general purpose tool for parallel deterministic systems, with applications to simulating cellular systems. *Journal of Physiology (London)*, 467:145, 1993.

[6] M Courtemanche, W E Skaggs, and A T Winfree. Two-dimensional rotating depolarization waves in a modified Beeler-Reuter model of cardiac cell activity. In G Cook, editor, *Science at the John von Neumann National Supercomputer Centre*, volume 3, pages 79–86. Consortium for Scientific Computing, Princeton, 1990.

[7] R FitzHugh. Impulses and physiological states in models of nerve membrane. *Biophysics Journal*, 1:445–466, 1961.

[8] M Gerhardt, H Schuster, and J J Tyson. A cellular automaton model of excitable media including curvature and dispersion. *Science*, 247:1563–1566, March 1990.

[9] K M Hobley, B C Thompson, and J V Tucker. Specification and verification of synchronous concurrent algorithms: a case study of a convolution algorithm. In G Milne, editor, *The Fusion of Hardware Design and Verification*, pages 347–374. North-Holland, 1988.

[10] A V Holden. Nonlinear science: the impact of biology. *Journal of the Franklin Institute*, 334B:971–1014, 1997.

[11] A V Holden. The restless heart of a spiral. *Nature*, 387:655–666, 1997.

[12] A V Holden. A last wave from the dying heart. *Nature*, 392:20–21, 1998.

[13] A V Holden, M J Poole, and J V Tucker. Reconstructing the heart. *Chaos, Solitons and Fractals*, 5:691–704, 1995.

[14] A V Holden, M J Poole, and J V Tucker. An algorithmic model of the mammalian heart: propagation, vulnerability, re-entry and fibrillation. *International Journal of Bifurcation and Chaos*, 6:1623–1635, 1996.

[15] A V Holden, M J Poole, and J V Tucker. A theoretical framework for analysis and synthesis of networks of neurones. In R Moreno-Diíaz and J Mira-Mira, editors, *Brain Processes, Theories and Models*, pages 320–329. MIT Press, 1996.

[16] A V Holden, M J Poole, J V Tucker, and H Zhang. Coupling CMLs and the synchronisation of a multilayer neural computing system. *Chaos, Solitons and Fractals*, 4:2249–2268, 1994.

[17] A V Holden, J V Tucker, and B C Thompson. Can excitable media be considered as computational systems? *Physica D*, 49:240–246, 1991.

[18] A V Holden, J V Tucker, H Zhang, and M J Poole. Coupled map lattices as computational systems. *Chaos*, 2:367–376, 1992.

[19] J P Keener. Propagation and its failure in coupled systems of discrete excitable cells. *SIAM Journal of Applied Mathematics*, 3:556–572, 1987.

[20] A R Martin and J V Tucker. The concurrent assignment representation of synchronous systems. *Parallel Computing*, 9:227–256, 1988.

[21] K Meinke and J V Tucker. Specification and representation of synchronous concurrent algorithms. In F H Vogt, editor, *Concurrency '88*, Lecture Notes in Computer Science 335, pages 163–180. Springer-Verlag, 1988.

[22] P M F Nielsen, L J Le Grice, B H Smaill, and P J Hunter. Mathematical model of geometry and fibrous structure of the heart. *American Journal of Physiology*, 260:H1365–H1378, 1991.

[23] D Noble. Oxsoft Heart program, Oxsoft Ltd, Oxford, 1990.

[24] A V Panfilov and A V Holden. Spatiotemporal irregularity in a two-dimensional model of cardiac tissue. *International Journal of Bifurcation and Chaos*, 1:219–225, 1991.

[25] A V Panfilov and A V Holden. Computer simulation of re-entry sources in myocardium in two and three dimensions. *Journal of Theoretical Biology*, 161:271–285, 1993.

[26] A V Panfilov and A V Holden, editors. *Computational Biology of the Heart*. John Wiley, 1997.

[27] M J Poole, J V Tucker, and A V Holden. Hierarchies of spatially extended systems and synchronous concurrent algorithms. In B Möller and J V Tucker, editors, *Prospects for Hardware Foundations*. Springer-Verlag, 1998. To appear.

[28] J J Rice, R W Winslow, and P Kohl. Conduction velocity in a model strand of cardiomyocites is linearly related to intracellular sodium concentration. *Journal of Physiology (London)*, 487:15–16, 1995.

[29] B C Thompson and J V Tucker. Equational specification of synchronous concurrent algorithms and architectures (2nd edition). Technical Report CSR 15-94, Department of Computer Science, University of Wales Swansea, 1994.

[30] J V Tucker. Theory of computation and specification over abstract data types and its applications. In F L Bauer, editor, *Logic, Algebra and Computation*, pages 1–39. Springer-Verlag, Berlin, 1991.

[31] J V Tucker and J I Zucker. Theory of computation over stream algebras, and its applications. In I M Havel and V Koubek, editors, *Mathematical Foundations of Computer Science*, Lecture Notes In Computer Science 629. Springer-Verlag, 1992.

[32] J V Tucker and J I Zucker. Computable functions on stream algebras. In H Schwichtenberg, editor, *Proof and Computation*, pages 341–382. Springer-Verlag, 1994.

[33] R L Winslow, D Cai, A Varghese, and Y-C Lai. Generation and propagation in models of cardiac sinus node and atrium. This volume, 1993.

[34] R L Winslow, A L Kimball, A Varghese, and D Noble. Simulating cardiac sinus and atrial network dynamics on the Connection Machine. *Physica D*, 64:282–298, 1993.