Computational multiscale modeling in the IUPS Physiome Project: Modeling cardiac electromechanics

D. Nickerson
M. Nash
P. Nielsen
N. Smith
P. Hunter

We present a computational modeling and numerical simulation framework that enables the integration of multiple physics and spatiotemporal scales in models of physiological systems. This framework is the foundation of the IUPS (International Union of Physiological Sciences) Physiome Project. One novel aspect is the use of CellML, an annotated mathematical representation language, to specify model- and simulation-specific equations. Models of cardiac electromechanics at the cellular, tissue, and organ spatial scales are outlined to illustrate the development and implementation of the framework. We quantify the computational demands of performing simulations using such models and compare models of differing biophysical detail. Applications to other physiological systems are also discussed.

Introduction

Mathematical modeling may be used to integrate data from electrical and mechanical cardiac experiments in order to test hypotheses that are concerned with multiple spatiotemporal scales and functions. This kind of integration across a variety of size and time scales is among the most advanced examples of organ system modeling [1–3]. A goal of the IUPS (International Union of Physiological Sciences, *www.iups.org*) Physiome Project (*www.physiome.org.nz*) is to develop the technology and methods required to simulate the behavior of biological organisms via numerical simulations using computational and mathematical models. Such simulations require the integration of multiple types of physics over a wide variety of spatial and temporal ranges, for example, spatial scales of 10^{-9} meters for subcellular structures up to approximately one meter for the human body, and molecular events occurring on the 10^{-6}-second time scale up to the human lifetime of the order of 10^9 seconds (**Figure 1**).

As the complexity of computational models increases, formal vocabularies are needed to reduce the growing heterogeneity of biological and mathematical expressions. Standards are being developed to formalize the description of experimental data and mathematical models of physiological processes [4, 5]. Ontologies that incorporate semantic descriptions of modeling concepts eliminate ambiguities in the modeling environment. To this end, ontologies and representation languages that include these ontologies are being developed under the IUPS Physiome Project, which facilitates communication of biological models to researchers through tools for building, sharing, interpreting, and visualizing models [6].

With the development of these languages comes the ability to create and populate repositories of models that are freely available for use by the scientific community. Currently, the most advanced of the "physiome" representation languages is CellML, a markup language based on open standards. Initially designed for application to models of cellular electrophysiology and reaction pathway models, CellML has since been used in a wide range of mathematical models, including constitutive laws for continuum mechanics. The term *physiome* refers to the quantitative and integrated description of the functional behavior of the physiological state of an individual or species.

In this paper, we review the development of a computational modeling framework that enables scientists to perform numerical simulations using models of integrative physiological function across the cellular, tissue, and organ spatial scales. The framework was initially developed with a focus on overcoming the problem of modeling the tightly coupled electromechanical function of the heart, but the

0018-8646/06/$5.00 © 2006 IBM

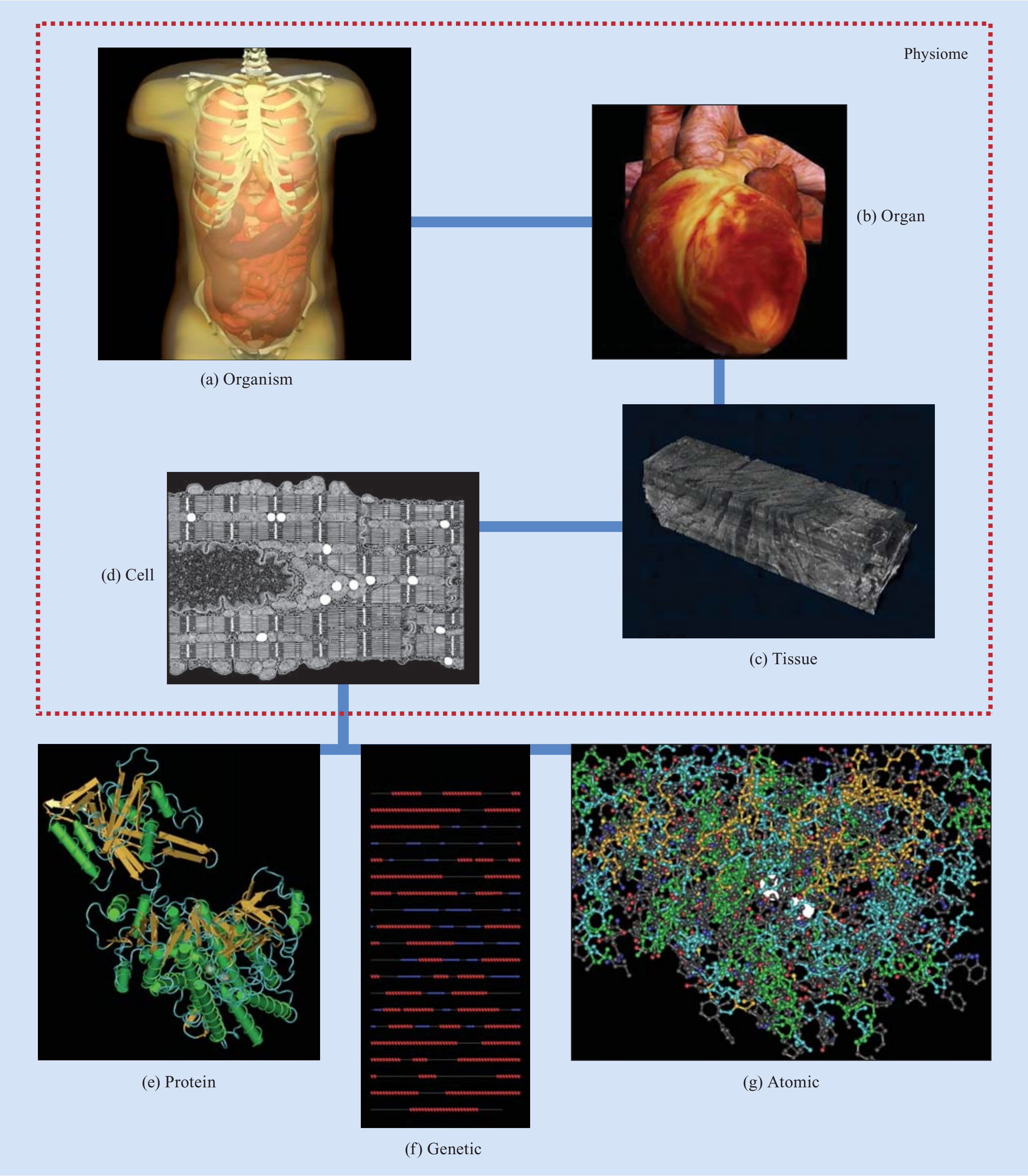


Figure 1

Hierarchy of spatial scales used in the IUPS Physiome Project, which concerns elements in the dashed box. Below the dashed box are protein and molecular size scales. (a) The current Auckland virtual human torso model. (b) Textured virtual heart. (c) Volume rendering of a piece of tissue removed from the left ventricular free wall of a rat heart. (d) Diagram of an idealized cardiac muscle cell based on electron microscope images. (e) Structural representation of the cardiac sarcoplasmic-reticulum calcium ATPase protein with two bound calcium atoms. (f) Amino-acid sequence for the protein in (e). (g) Detailed view of the atomic structure of the protein in (e), with the two calcium atoms shown in white.

implementation is sufficiently general that applications to other areas of modeling are underway.

The framework we have developed allows the use of CellML for the specification of model- and simulation-specific mathematical equations—for example, cellular electrophysiology models and passive mechanical functions. The simulation framework is built upon the extensive base provided by the CMISS software package developed at the Bioengineering Institute at The University of Auckland, New Zealand. CMISS is an interactive computer program for Continuum Mechanics, Image analysis, Signal processing and System identification. It is also a modeling environment that allows the application of several mathematical techniques to a variety of complex bioengineering problems.

CellML

CellML (*www.cellml.org*) is an XML-based language developed by the Bioengineering Institute at The University of Auckland [4] (originally in collaboration with Physiome Sciences Inc., but now entirely supported by New Zealand's Public Good Science Funding). CellML is a language designed to store and exchange computer-based biological models; where appropriate, the language builds upon existing XML standards such as MathML (*www.w3.org/Math*) for the specification of mathematical equations and the Resource Description Framework (*www.w3.org/RDF*) for the encapsulation of metadata.

CellML provides a relatively basic set of tags that can be used to mark up complex interactions between a set of mathematical equations represented in the MathML language. See **Figure 2** for an example of CellML code. Although CellML clearly provides a means for making models available to researchers for validation and study, these models must also be published and peer-reviewed before being accepted by the modeling community [7].

Through the use of CellML and the open-source tools that are now becoming available (*cellml.sourceforge.net*), an author of a model is able to describe a model in such a way that others are able to incorporate it into their own computational codes or simulation package of choice. The model equations can be specified just once and used in all implementations and publications of the model. Similarly, all boundary and initial conditions required for a particular computational experiment can be specified just once and used by the community. With the establishment of a publicly accessible repository of models and simulation tools, authors are able to submit validated models and simulation results to ensure that other investigators are able to accurately reproduce their simulations. In this paradigm, the onus of model validity no longer rests with the model user, but with the model author and the software engineers who implement the CellML application libraries and tools. The repository of models and simulation results consists of data that the author asserts is an accurate representation of a model; it provides a much larger basis for testing the code and ensuring robustness and compatibility.

```
<component name="IbCa">
     <variable name="IbCa" units="pA_per_pF"
public_interface="out"/>
     <variable name="g_bCa" units="nS_per_pF"
        public_interface="in"/>
     <variable name="Vm" units="mV"
        public_interface="in"/>
     <variable name="E_Ca" units="mV"
        public_interface="in"/>
     <math xmlns="http://www.w3.org/1998/Math/MathML">
        <apply id="i_bCa_calculation"><eq/>
              <ci>IbCa</ci>
              <apply><times/>
                   <ci>g_bCa</ci>
                   <apply><minus/>
                        <ci>V</ci>
                        <ci>E_Ca</ci>
                   </apply>
              </apply>
        </apply>
     </math>
</component>
```

Figure 2

Sample piece of CellML code representing the background calcium current found in cardiac cells and defined by the equation $I_{\mathrm{bCa}} = g_{\mathrm{bCa}}\,(V_{\mathrm{m}} - E_{\mathrm{Ca}})$. In this component (the smallest functional unit in a CellML model), we define the current variable `IbCa` as an output, and the parameters of the equation (conductance `g_bCa`, membrane potential `Vm`, and reversal potential `E_Ca`) as inputs. The subscript "b" stands for background, and g is the variable for conductance.

As model repositories are being developed, researchers need to be confident that a given representation or implementation of a model is "accurate"—that is, the degree to which the representation or implementation reflects the reference description of the model, or the accuracy with which the underlying phenomena represented by the model are reproduced. Standards are being developed that will address these concerns [8] and will be incorporated into CellML models through the use of curation metadata that will be crucial for the general acceptance and use of a CellML model repository.

The CellML model repository contains models from peer-reviewed journal publications. When researchers make a CellML version of a published model available in the repository, this does not guarantee that the model is error-free, because the original publication may have errors, and the corresponding CellML model versions, for issues of provenance, must faithfully reproduce the mathematics in the paper, errors included. However, it is clearly desirable to create a second version of the CellMLencoded model in which various checks have been carried out, including 1) checking that the defined units of measurement are consistent; 2) checking that all parameters and initial conditions are defined; and ultimately 3) checking that running the model reproduces the results published in the paper. As the CellML standard and the model repository gain acceptance by journals, it should be possible to work with authors to achieve this refined version of the model at the time of publication. For example, such refinement has happened during publication of a metabolic model [9]. A further level of model curation is anticipated in which models are checked for the extent to which they satisfy physical constraints such as conservation of mass, momentum, and charge in chemical and physical interactions.

Owing to the completely generic specification of the language, CellML has a much broader range of applicability than the name suggests. CellML is considered to be a language suitable for the description of annotated mathematics, i.e., the specification of mathematical equations using MathML, the interrelationships between equations, and the connections between variables contained in the equations. Model authors and users are further able to annotate models by using metadata associated with any data contained in the model.

Other XML-based languages have been developed to aid the exchange of mathematical models within various scientific communities. These languages have tended to be tied to specific domains in terms of their use and the actual definition of the language syntax. For example, CellML is often compared to the Systems Biology Markup Language (SBML), which is currently specifically for use with biochemical network models, although future development plans for the language begin to introduce concepts similar to those in CellML. A particularly powerful feature introduced in CellML 1.1 is the ability to reuse models, or parts of models, by "importing" the relevant mathematical equations and variable definitions into a new model. As we have suggested, the generic syntax used in the CellML language implies no specific domain of application, which allows its use in a wide variety of scenarios that allow model authors to annotate models with domain-specific metadata.

CMISS

As mentioned, CMISS (*www.cmiss.org*) is a computational package for modeling the structure and function of biological systems. In particular, it is designed to model the anatomy and behavior of organ *systems* (e.g., cardiovascular, respiratory, and special sense organs) from the component organs (e.g., heart, lungs, and eyes), while also considering the cellular and subcellular scales and the coupling that occurs between and within all of these levels. Equations derived from physical laws of conservation, such as conservation of mass, momentum, and charge, are solved in order to predict the integrative behavior of an organ, given descriptions of the anatomical structure and tissue properties. The tissue properties used in these organ simulations can incorporate tissue structure and cellular processes, together with spatial variation of the parameters, such as parameters associated with mechanical compliance and electrical conductivity, that characterize these processes. CMISS has facilities for fitting models to geometric data derived from various imaging modalities (e.g., MRI, CT, and ultrasound) and has a rich set of tools that permit graphical interaction with the models and display of simulation results. A graphical user interface, an interactive console-based interpreter, or batch-mode scripting may be used to control CMISS.

CMISS originated in the doctoral work of Peter Hunter [10] as a finite-element program for stress analysis of large deformations in the heart. The package has since evolved into a general-purpose biological-systems modeling tool, used in the areas of continuum mechanics, image analysis, signal processing, and system identification. Recently, work has begun to modularize CMISS in order to enable the development of specialized and focused tools for various medical and other applications. The main academic goal of CMISS is to support the IUPS Physiome Project. See [6] for a detailed review of the abilities of CMISS in relation to the heart.

As the Physiome Project evolves, it is essential to provide both programmer and user access to the technologies being developed. For example, application developers must be able to access the model repositories and the data contained within model-representation documents, while users must be able to interact with specific models and perform simulations. To achieve this support, the user interface of CMISS is separated from the main application and is released under an open-source license. As part of this software evolution, standard application program interfaces (APIs) are being developed for various components of the software, which is being divided into separate modules that can be linked into external applications.

With the development of these modules, customized user interfaces can be created for specific modeling or simulation platforms along with interfaces capable of browsing model repositories. To aid in the sharing of information, most of these interfaces will be "Web-deliverable" either through the use of standard web technology or via a Mozilla extension that implements the APIs of the open-source CMISS modules. Mozilla extensions are applications that can be added into an existing Mozilla-based web browser (e.g., Firefox** or Mozilla) to provide extended functionality. Using this extension, graphical interfaces to Physiome technology can be specified using the XML user interface language (XUL) and delivered via the Internet to any XUL-enabled client such as the Firefox web browser.

Incorporation of CellML into CMISS

Our goal was to implement a process for enabling the specification of model-specific mathematical equations in CMISS through CellML. For example, prior to this work, a new cellular model in CMISS was implemented by writing the equations in Fortran and adding this to the CMISS code base. This method has two main disadvantages: The author of the Fortran code is usually translating a published piece of work, which can lead to human errors in the translation, and the generated code is very specific to CMISS and thus not easily transferred to another modeling or simulation package. This also assumes that the published model itself is free from errors. Like problems associated with cellular models, the hard-coding of other mathematical equations into the CMISS code base has disadvantages.

To avoid these problems, CMISS is capable of importing mathematical models from CellML (**Figure 3**). This provides the ability to store and simulate models in an open standard, even though the models may conceivably originate from various sources.

The first step in implementing this ability in CMISS required us to develop a standard API for use when accessing a CellML model description. The most recent API implementations are freely available from the CellML website, *cellml.sourceforge.net*. With an API defined, the ability to import CellML was added to CMISS, allowing the definition of mathematical models via CellML. This also required the capability of translating the mathematical expressions from the CellML model (stored as MathML) into a dynamically loadable object that can be utilized by CMISS during a given simulation. Given the structure defined by MathML, the translation to such an object is reasonably straightforward (**Figure 4**).

Using the described approach, a scientist can develop a cellular model using software that is well suited to single-cell modeling or that may be used in experimental work. If the software is capable of exporting CellML, it is possible to use a cellular model and perform tissue and whole heart simulations that are based on the model.

In order to import CellML models into tissue representations, we must be able to specify spatial variations of these models and their parameters within the larger-scale model. For example, when modeling the spread of excitation from the pacemaker cells in the sino-atrial node into the atria, a modeler would typically use

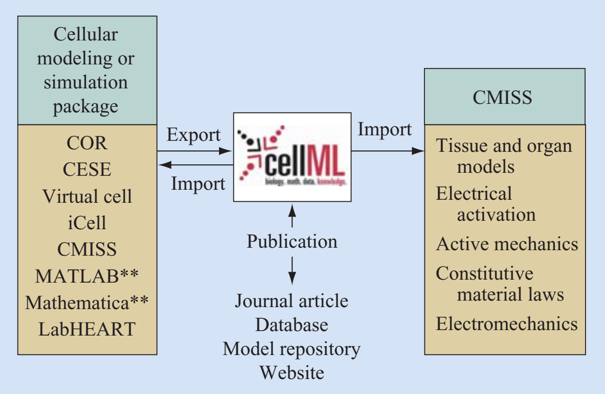
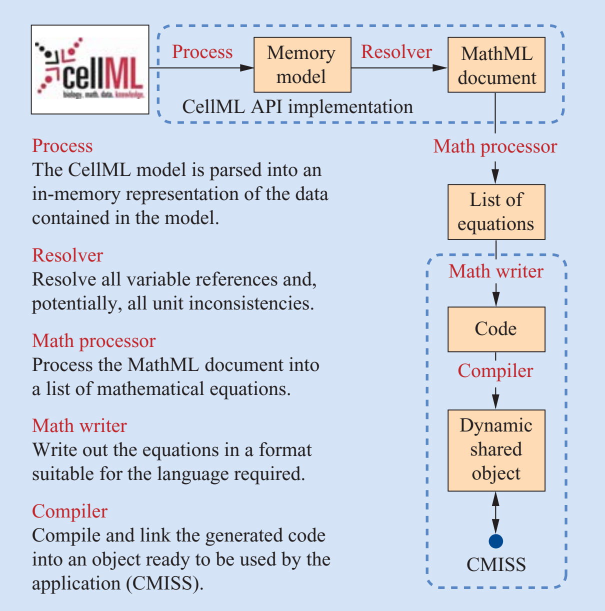


Figure 3

Illustration of the way CellML can be used to facilitate the development of mathematical models using domain-specific software, while allowing the models to be easily incorporated into tissue- and organ-level models. The cellular modeling and simulation packages listed at left are examples of software that currently or soon will have the ability to read CellML models.



Process
The CellML model is parsed into an in-memory representation of the data contained in the model.

Resolver
Resolve all variable references and, potentially, all unit inconsistencies.

Math processor
Process the MathML document into a list of mathematical equations.

Math writer
Write out the equations in a format suitable for the language required.

Compiler
Compile and link the generated code into an object ready to be used by the application (CMISS).

Figure 4

Illustration of the work flow involved in the generation of code suitable for use in CMISS from a CellML source. The upper dashed box encapsulates the processes that are internal to the CellML API implementation, and the lower encapsulates those internal to CMISS. The Math processor is independent and external to both of these. The Math processor may require simplification of the equations. The Math writer may also require the appropriate sorting of equations for the language being used. For the Process part, the model may come from an XML file, through a connection to a database or through another application.

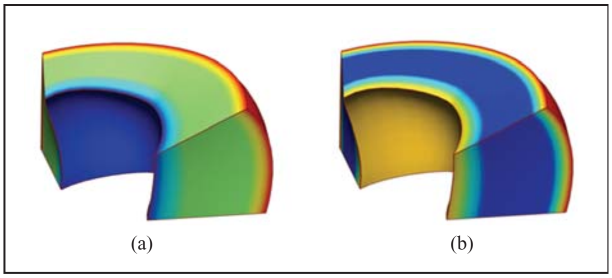


Figure 5

Examples showing the variation of the g_{to} and g_{Ks}/g_{Kr} parameters through the ventricular wall. In (a), g_{to} is 0.0005 mS $\cdot$ mm^{-2} at the endocardial surface (blue), 0.005 mS $\cdot$ mm^{-2} in the midmyocardium, and 0.011 mS $\cdot$ mm^{-2} at the epicardial surface (red). In (b), g_{Ks}/g_{Kr} is 19 at the endocardial surface (yellow), 7 in the midmyocardium, and 23 at the epicardial surface (red). The geometry is a wedge taken from a left-ventricle wall of the porcine ventricular model, and the parameter variation is described in [26]. Variables used: g, membrane conductance; g_{Ks}, g_{Kr}, conductances for the slow and rapid potassium currents; *to*, transient outward current.

different cellular electrophysiological models for each of these two regions of tissue. The variation of channel distributions through the ventricular wall of the heart (**Figure 5**) is another example of the need for spatial variation. Here, the modeler may want the same equations represented at all points in the ventricles, but needs to specify spatial distributions of channel densities.

Multiscale modeling of cardiac electromechanics

The modeling framework that we have developed uses cellular models of cardiac electromechanics to drive the dynamic functional behavior of tissue and organ continuum models. Electrical excitation and mechanical deformation at the tissue and organ spatial scales, in turn, modulate changes in cellular model material properties.

Cardiac cellular electrophysiological modeling is a well-established field with numerous existing models that cover a large range of species and cellular phenotypes (see [7, 11, 12] for recent reviews). Similarly, the mechanical behavior of cardiac cells has been extensively investigated through the use of mathematical models. In recent years, modeling of cellular metabolism and energetics [13, 14] has been considered an established field, and this modeling has gained prominence because of its relevance to many dysfunctions of the heart. Each of these functions is closely integrated within a cardiac cell, and some models have been developed that reflect this tight coupling at the cellular or subcellular level [13, 15]. Another approach requires the development of methods that enable independent models to be coupled together, and this approach has been shown to work particularly well when coupling cellular electrophysiological and mechanical models [16].

Cellular models have been developed with varying levels of biophysical detail. For example, when modeling various phenomena, different approximations can be made to simplify the cellular model in order to dramatically decrease the computational demands required for simulations using these models. Thus, cellular models are typically identified as either “low-dimensional” or “biophysically based” models. Low-dimensional models are used to represent gross behavior of the system, sometimes based on the use of mathematical analysis to reduce complex models, given certain constraints. A classical low-dimensional model of the cellular action potential is the FitzHugh–Nagumo model [17, 18], based on a phase-plane analysis of the Hodgkin–Huxley nerve axon model [19] and modified in several cardiac-specific action potential models [20, 21].

In contrast to the relatively simple equations of low-dimensional models, biophysical cellular models attempt to accurately represent detailed physiological processes and mechanisms that underlie the phenomena being modeled. In the case of cardiac electrophysiology at the whole-cell spatial scale, this includes the dynamics of various ionic species and the gating kinetics of various proteins to permit or block the transport of ions between distinct compartments. Such models generally consist of large systems of stiff ordinary differential equations. Models based on the work of C. Luo and Y. Rudy [22, 23] and D. Noble [24] have been widely used and adapted to various specific situations. As the quantity and quality of experimental data and techniques improve, greater biophysical detail can be extracted. Furthermore, models are now being developed to reproduce the effects of genetic mutations that govern the dynamics of specific transmembrane ion channels. These models are beginning to incorporate more realistic stochastic behavior of protein populations contained in single cells or populations of entire cells, leading to significant increases in the complexity of the models [15, 25, 26].

Modeling cardiac electromechanics on a spatial scale larger than for the single-cell models described above requires the coupling of two processes: the spread of electrical excitation through the tissue and the mechanical response of the tissue. In a normal mammalian heart, the spreading wave of electrical excitation triggers contraction of the cardiac muscle, which is responsible for the pumping blood. While the aforementioned two processes can be modeled independently, it is well established that mechanisms of excitation–contraction coupling and mechano-electrical feedback are tightly linked [27].

A large body of research exists for modeling the spread of electrical excitation throughout cardiac tissue as well as the whole heart [12, 28]. Continuum models are based on the assumption that the length scales of the physically observable phenomena are large in comparison to the underlying discrete structure of the material. Our group has developed numerical simulation techniques for solving continuum models of electrical excitation, based either on the bidomain model (i.e., a model involving solving for intra- and extra-cellular potential fields, assuming two interpenetrating domains) [29, 30] or an eikonal-type model for activation times, which involves instantaneous solving for electrical activation times throughout the solution domain [31].

Although we have the tools to model the full bidomain model, in this work we have reduced the complexity of the simulations by using the simplified monodomain model [32] and neglecting the effect of extracellular potential on the electromechanical behavior of the tissue. These assumptions are valid when simulating the normal spread of electrical excitation, but the full bidomain model would be required in other circumstances, such as when defibrillation shocks are being applied [33]. Complexity reduction allows a significant saving in terms of computational cost in both memory requirements and solution times.

Similarly, a large body of research also exists for modeling the mechanical behavior of cardiac tissue (see [28] and [34] for reviews). In the past, many models of cardiac mechanics considered only the passive properties of the muscle for reasons of model complexity and the lack of experimental data during the systolic phase of the cardiac cycle (i.e., during the contraction of isolated tissue preparations). This allowed quasi-static models of finite-deformation elasticity to be applied to the heart with great success. Finite-element continuum models of cardiac mechanics are the most prevalent in this field, and the use of high-order interpolation of fields, in which our group specializes, is well suited to these types of models.

The development of models of electrophysiology and mechanics has largely occurred independently, and only recently have we begun to obtain the computational power and experimental data required to develop models of electromechanics in cardiac tissue. Various approaches have been used in the development of cardiac electromechanics models providing varying levels of physiological detail and interactions between electrical and mechanical processes. Such tissue models consider tight interaction between mechanics and electrophysiology using low-dimensional cellular models [35–37], and also models with less interaction between the mechanics and electrophysiology during a simulation (e.g., either excitation–contraction coupling or mechano-electrical feedback). These latter models are based on more biophysically detailed cellular models (e.g., [38–40]). In this case, models have typically solved for electrical activation times and used these times to trigger local active contraction of the cardiac tissue. The activation times can be computed either from a simulation of electrical activation or through the use of an eikonal model to solve for activation times directly [31]. In these loosely coupled frameworks, the spread of electrical activation is calculated independently of both mechanical deformation and mechano-electrical feedback mechanisms such as stretch-activated channels and calcium buffering by contractile proteins.

In the work described in the current paper, we have used a large-scale, high-order-interpolation, finite-element-based method for solving mechanics, coupled to a small-scale, low-order interpolation method for solving electrical activation in order to produce a technique for the numerical solution of biophysically detailed cardiac electromechanics models [41]. Our tightly coupled framework uses cellular models of electromechanics to drive the dynamic functional behavior of the model, while the properties of the cellular models are modulated by both the electrical excitation and the deforming mechanical model.

Owing to the computational resources required for modeling three-dimensional electromechanics, we have been limited to simulations that use either complex cellular models with simplified (small) geometrical models or low-dimensional cellular models with more anatomically based (large) geometries. Here we present results obtained from a cube of tissue, and some preliminary results from a simplified geometric model of the cardiac left ventricle, as an illustration of the application of the framework discussed above. Further analysis and discussion of the left-ventricular results can be found elsewhere [41].

Results

We first present simulation results from the cube shown in **Figure 6**. We performed simulations to investigate the response of this tissue block to the applied electrical stimulus using two cellular electromechanics models: a low-dimensional and a biophysical model. The low-dimensional model was obtained through the coupling of the Fenton–Karma (FK) cardiac action potential model [42] to the Hunter–McCulloch–ter Keurs (HMT) mechanics model [43], following the procedure described in [16]. For the biophysical model, we coupled the HMT mechanics model to the most recent of the Luo–Rudy-based models developed to investigate various genetic mutations [25, 26, 44], with the addition of a more biophysically detailed model of calcium dynamics [45]. In the following, we denote the low-dimensional model as the FK–HMT model and the biophysical model as the N–LRd–HMT model.

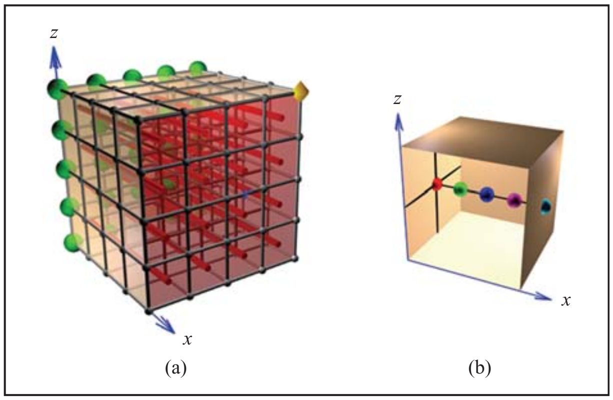


Figure 6

(a) Geometry and boundary conditions used for the cardiac cube model. The overall cube measures $4 \times 4 \times 4$ mm, consisting of 64 unit cubes. The nodes represented by green spheres are fixed in the y–z plane and that represented by the gold diamond is restricted to slide along the x-axis. The solid cylinders within the cube indicate the fiber orientation. An electrical stimulus is applied on the $x = 4$ face, indicated by the red face of the cube. (b) Spatial locations for the cellular transients presented in Figure 7.

Figure 7 illustrates a summary of the results from the biophysical simulation using the N–LRd–HMT cellular model for the tissue cube shown in Figure 6. In Figures 7(a) and 7(b), the wave of electrical activation can be seen advancing through the cube from the stimulus (isoelectric surfaces are shown in color, with blue and red shading respectively representing the most negative and the most positive potentials). Following activation, the adjacent tissue contracts along the fiber direction [shown in Figure 6(a)] due to the dynamic development of tension within the cells triggered by the electrical excitation (the axis of cardiac cells is aligned with the fiber direction). Given the incompressible nature of cardiac tissue, expansion occurs in the other two dimensions. Following excitation, the tissue recovers and returns to the initial resting state.

Figure 7(c) shows key transients from the cellular models at the spatial locations within the tissue cube shown in Figure 6(b). The transients shown include the cellular action potentials, dynamic tension, and a measure of cellular length. These transients show the smooth propagation of electrical excitation through the tissue, from the cyan point on the stimulated face to the red point farthest from the stimulus. Following a time delay (where electrical excitation triggers internal cellular processes that give rise to tension generation), the dynamic tensions of the cells begin to rise, causing the cells to shorten and hence causing the tissue to contract. Also illustrated is the initial stretch of tissue in regions distal to the stimulus due to the passive cells being stretched as the actively contracting tissue acts against the applied displacement boundary conditions, as described in Figure 6. As tissue is activated, it follows a similar contraction transient.

The dip into negative tension shown in Figure 7(c) was unexpected and was initially thought to result from poor numerical convergence. Further investigation revealed that this was not the case, and that the negative dip could be attributed to mismatched material parameters and deficiencies in the model. Material parameters for the cellular, electrical activation, and finite elasticity components of the model have been taken from previous studies in which the parameter values of each component have largely been established independently. Further material parameter estimation studies are required to determine a more appropriate set of parameters in order to better match experimental observations. Such studies are currently expensive to perform because of the computational requirements of these simulations (**Figure 8**). In addition to material parameter mismatches, acknowledged deficiencies exist in the active contraction model that contribute to non-physiological behavior in both the initial phase of the tension transient and during relaxation. These deficiencies have been addressed at the cellular level in a recent study [46].

Both the FK–HMT and N–LRd–HMT cube models consist of 64 high-order finite elements used for the solution of the equations of finite deformation elasticity and 46,656 low-order finite elements for the electrical propagation solution representing 35,937 cells at a spacing of 0.125 mm. The simulations were performed on an IBM eServer* p690 computer (frequently referred to as "Regatta") with 32 1.3-GHz POWER4* processors and 32 GB of main memory.

The simulation using the FK–HMT cellular model required approximately 550 MB of memory while the N–LRd–HMT model required 1.5 GB, which will be of concern for model extension to more realistic geometries that require millions of cells. CPU and wall clock times for these simulations are summarized in Figure 8, with the simulations performed using eight processors and the shared-memory parallel implementation of CMISS. In Figure 8, the simulation time for each of the models is split into the three main components: solution of the electrical activation model, solution of the finite elasticity mechanics model, and the update step transferring data between the two models to integrate the feedback effects.

As shown in Figure 8, the main difference in solution time between the FK–HMT and N–LRd–HMT models is due to the electrical propagation model. In this step, the full cellular model is integrated over a time step at each of the 35,937 cells in the tissue cube, and the extra complexity in the biophysical N–LRd–HMT cellular model significantly increases the computational

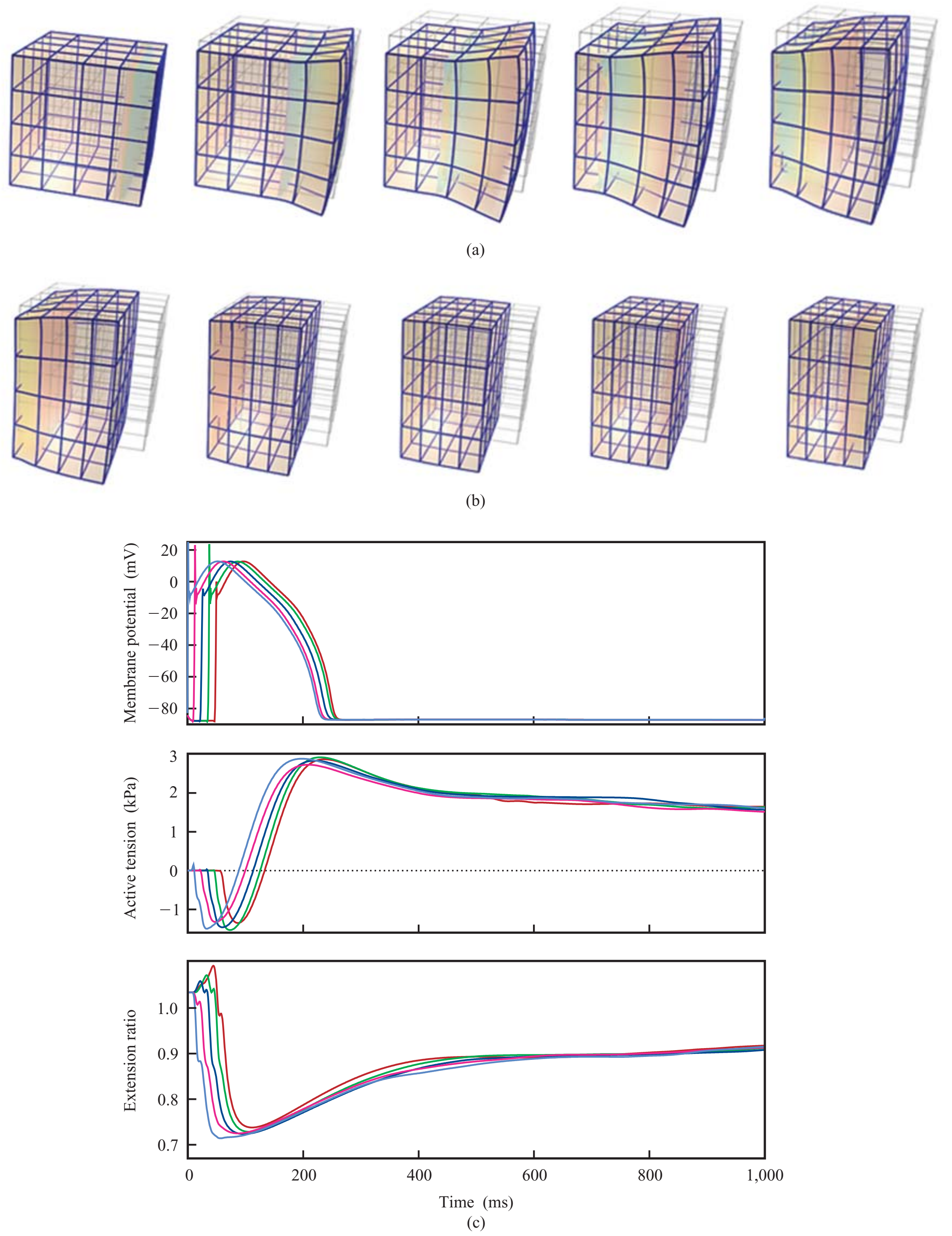


Figure 7

Simulation results from the N–LRd–HMT electromechanics cube model. Parts (a) and (b) show deformation solutions in 10-ms time steps with transmembrane potential isosurfaces (red most positive and blue most negative). Part (a) corresponds to the time range of 10 to 50 ms. Part (b) continues the sequence in (a) and corresponds to a 60 to 100 ms time range. Part (c) shows graphs of the three cellular parameters at the spatial locations indicated by the matching color spheres in Figure 6(b) as a function of time.

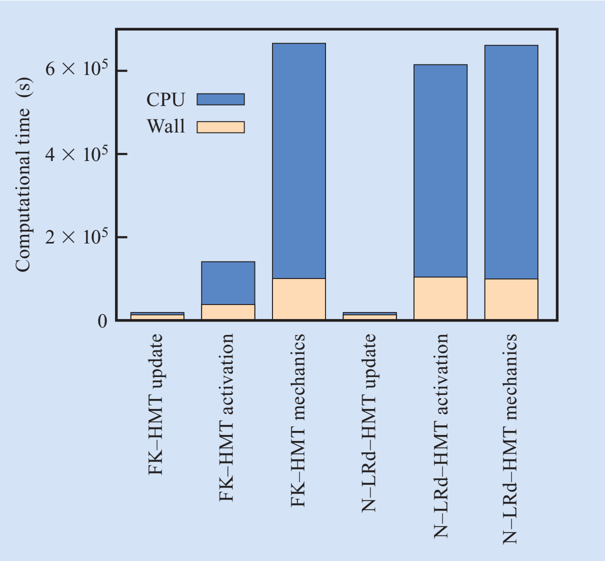


Figure 8

Comparison of the total computational times for the FK–HMT and N–LRd–HMT cube electromechanics models. CPU time is the total computational time required across all processors for the simulations. Wall clock time is the actual duration of the simulations. The ratio of the wall clock to CPU times provides an indication of the speedup obtained through the use of the multiprocessor machine.

time required for model solution. During the mechanics solution step, no temporal integration is required in order to update the steady-state active tension solution, so we simply evaluate the algebraic expressions defined in the HMT model. If we were to use a more numerically complex mechanics model (see [15] for an example), we would expect an increase in computational time for the mechanics solution step similar to that shown in Figure 8 between the activation solution steps for the FK–HMT model and those for the N–LRd–HMT model. The time-dependent, dynamic component of active force development is included in the starting solution of each mechanics time step, but we neglect changes in the dynamic component during the application of numerical length perturbations in order to facilitate the numerical techniques being used to solve the finite elasticity model.

In summary, the relatively simple FK–HMT-based model of a 4 × 4 × 4-mm tissue cube required 550 MB of memory and 43 hours of compute time (wall clock) to perform a one-second contraction simulation. The N–LRd–HMT-based model, which includes a biophysically detailed model of cellular electrophysiology, required 1.5 GB of memory and 61 hours of compute time. These numbers obviously have a serious potential impact when simulations on more complex geometries within a reasonable time period are under consideration. We have, however, recently extended this work to a simplified model of the left ventricle (LV) of the heart [41].

Figures 9 and **10** present the results of a simulation of the contractile portion of the cardiac cycle in a simple left-ventricular geometry. At the cellular level, this model consists of the FK–HMT electromechanics model described above. The pole-zero constitutive law [47] is used to describe the passive mechanical behavior of the tissue microstructure. A constant-volume cavity constraint provides the dynamic pressure load applied to the endocardial surface of the LV model during the isovolumic contraction and ejection phases of the cycle. During ejection, the cavity model extends beyond the basal plane of the LV endocardial surface, corresponding to the quantity of blood exiting the ventricular cavity.

The model presented in Figures 9 and 10 consisted of only 320,000 cells, much fewer than would be required for a spatially converged solution of the electrical activation model, but adequate for this initial study. This simulation was performed on the same IBM p690 machine described above, and memory requirements peaked at 4 GB. The one-second simulation required 304 hours of compute time (wall clock) running on 12 processors.

From the comparison of the FK–HMT and N–LRd–HMT cube models above, we predict that models of the LV using the N–LRd–HMT model would require approximately 12 GB of memory and 430 hours of compute time. However, the more complex biophysical N–LRd–HMT model may be unsolvable on such a coarse discretization for the electrical activation model. As we move to more realistic geometrical models, the computational requirements would increase even further.

Discussion

We have developed a novel computational modeling framework for the simulation of cardiac electromechanics. We have shown how our framework can be used for simulations that use geometric models that vary from tissue-block models to ventricular models. This framework enables the development and testing of new hypotheses associated with ventricular pacing, myocardial ischemia, and defibrillation. The most significant drawback in the framework highlighted by the simulations presented above is the sheer amount of computational time required to obtain these results. This computational limitation currently precludes the embedding of detailed cellular models in more anatomically based geometries. By combining the N–LRd–HMT model with existing anatomically based models of the heart [48, 49], a model could be created that accounts for mechano-electrical feedback and stretch dependence in an anatomically based ventricular geometry via the tight-coupling solution procedure outlined above. This coupling between cellular

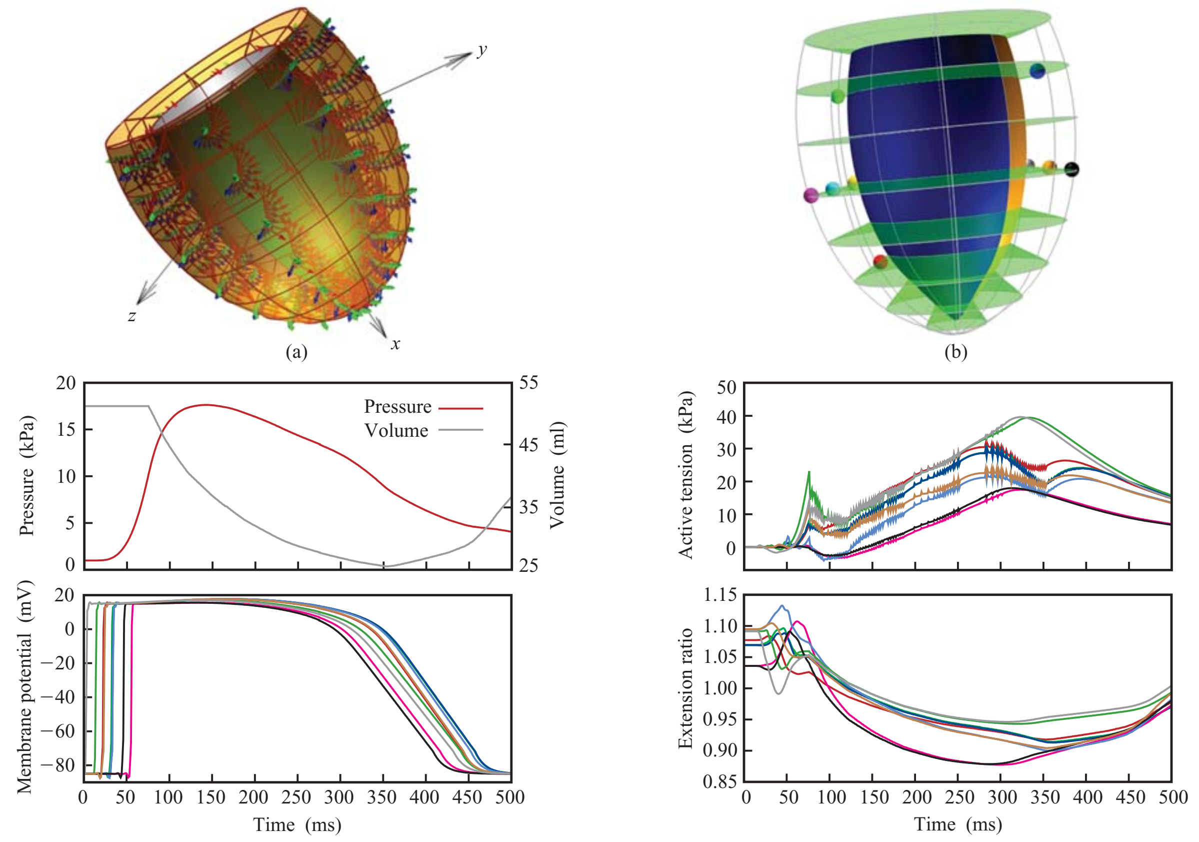


Figure 9

Active contraction and ejection of blood using a rotationally symmetric left-ventricular model. (a) Geometric model of the left ventricle with the underlying tissue microstructure represented by the colored arrows (red fiber axis, i.e., principal direction of cellular alignment; green sheet axis; and blue sheet-normal axis). (b) Spatial locations of the cellular transients shown below. The bottom four graphs show the cavity pressure and volume transients, cellular active tension, cellular membrane potential, and current cell length relative to resting length (i.e., extension ratio).

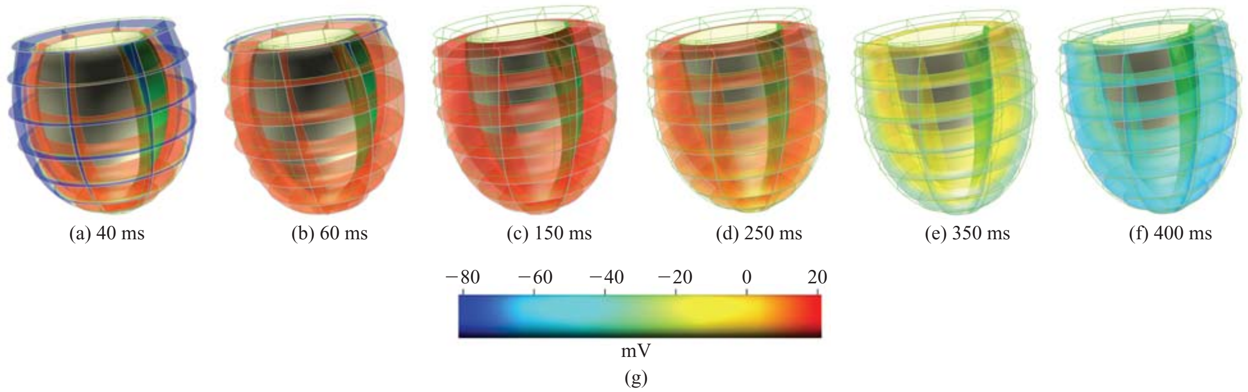


Figure 10

Simulation results of the active contraction and ejection of blood using the rotationally symmetric left-ventricular geometry shown in Figure 9(a). The green lines show the undeformed geometry and the colored surfaces indicate membrane electrical potential, using the color scale in (g).

contraction and activation has been proposed as a possible mechanism underpinning the heterogeneous electromechanical delay that is required to simultaneously produce physiological spatio-temporal sequences of both activation and contraction [39].

Our framework is sufficiently general that, through the use of CellML, new cell models can be accommodated without changing the existing software tools. Thus, as the rapid increase of computational resources continues, we anticipate that we will soon be using this framework with more detailed anatomy and biophysically based cell models. As an illustration of this, we performed the above cube simulations using a new IBM pSeries* 595 with 256 GB of memory and 64 1.9-GHz IBM POWER5* processors. The N–LRd–HMT-based simulation, which required 61 hours of compute time on the p690 machine using eight processors, ran in eleven hours using 32 processors on the p595. This shows the increased computational performance of the newer machine and provides the platform to perform more complex computations. In addition to simply obtaining larger and faster computers, we are also investigating many areas of algorithmic and software design that will provide even greater improvements in computational cost. For example, speed increases may be obtained by compilation of CellML models into optimized descriptions using lookup tables and partial evaluation, the use of multigrid techniques and adaptive local mesh refinement, and altering code design to make use of distributed massively parallel processing environments.

Further application of the research described above has also begun in other organ systems that involve electromechanics. For example, the musculoskeletal system is a large system in which modeling of electromechanics is important to the understanding of function. While the detail of the skeletal muscle electromechanics differs from that of cardiac muscle, the underlying methods we have developed for the cardiac models are equally applicable to skeletal muscle [50]. With CellML, it is straightforward to replace the cardiac cellular models with skeletal muscle models, and simulations of knee flexion, for example, have been performed [51].

Like skeletal muscle, smooth muscle undergoes active contraction in response to electrical stimuli. While skeletal and cardiac muscle involve somewhat similar time scales, smooth muscle is significantly slower. Again, the underlying modeling framework developed in this work is capable of representing models of contracting smooth muscle. Initial investigation of this class of muscles is currently underway in the modeling of an active bronchial airway [52]. The computational modeling framework discussed in this paper provides a simulation environment for the integration of various physics approaches over multiple space and time scales. As such, the framework provides an ideal platform for the development of innovative models of human physiology in order to aid medical sciences in clinical diagnosis and drug discovery.

Acknowledgments

This work was supported by the Wellcome Trust, the Royal Society of New Zealand (Centre for Molecular Biodiscovery), and the Marsden Fund.

*Trademark, service mark, or registered trademark of International Business Machines Corporation.

**Trademark, service mark, or registered trademark of The MathWorks, Inc., Wolfram Research, Inc., or the Mozilla Foundation in the United States, other countries, or both.

References

1. P. J. Hunter, P. Robbins, and D. Noble, "The IUPS Human Physiome Project," *Pflügers Arch.–Eur. J. Physiol.* **445**, No. 1, 1–9 (2002).
2. P. J. Hunter and T. Borg, "Integration from Proteins to Organs: The Physiome Project," *Nature Rev. Mol. Cell Biol.* **4**, No. 3, 237–243 (2003).
3. E. J. Crampin, M. Halstead, P. Hunter, P. Nielsen, D. Noble, N. Smith, and M. Tawhai, "Computational Physiology and the Physiome Project," *Exp. Physiol.* **89**, No. 1, 1–26 (2004).
4. A. A. Cuellar, C. M. Lloyd, P. F. Nielsen, D. P. Bullivant, D. P. Nickerson, and P. J. Hunter, "An Overview of CellML 1.1, a Biological Model Description Language," *Simulation* **79**, No. 12, 740–747 (2003).
5. M. Hucka, H. Bolouri, A. Finney, H. M. Sauro, J. C. Doyle, H. Kitano, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cuellar, S. Dronov, M. Ginkel, V. Gor, I. I. Goryanin, W. Hedley, T. C. Hodgman, P. J. Hunter, N. S. Juty, J. L. Kasberger, A. Kremling, U. Kummer, N. LeNovere, L. M. Loew, D. Lucio, P. Mendes, E. D. Mjolsness, Y. Nakayama, M. R. Nelson, P. Nielsen, T. Sakurada, J. C. Schaff, B. E. Shapiro, T. S. Shimizu, H. D. Spence, J. Stelling, K. Takahashi, M. Tomita, J. Wagner, and J. Wang, "The Systems Biology Markup Language (SBML): A Medium for Representation and Exchange of Biochemical Network Models," *Bioinformatics* **19**, No. 4, 524–531 (2003).
6. G. R. Christie, D. Bullivant, S. Blackett, and P. J. Hunter, "Modelling and Visualising the Heart," *Comput. Visual. Sci.* **4**, 227–235 (2002).
7. D. P. Nickerson and P. J. Hunter, "The Noble Cardiac Ventricular Electrophysiology Models in CellML," *Prog. Biophys. Mol. Biol.* **90**, No. 1–3, 346–359 (2006).
8. N. Le Novere, A. Finney, M. Hucka, U. Bhalla, F. Campagne, J. Collado-Vides, E. J. Crampin, M. Halstead, E. Klipp, P. Mendes, P. Nielsen, H. Sauro, B. Shapiro, J. L. Snoep, H. D. Spence, and B. L. Wanner, "Minimum Information Requested in the Annotation of Biological Models (MIRIAM)," *Nat. Biotechnol.* **23**, No. 12, 1509–1515 (2005).
9. D. A. Beard, "A Biophysical Model of the Mitochondrial Respiratory System and Oxidative Phosphorylation," *PLoS Comput. Biol.* **1**, No. 4, e36 (2005).
10. P. J. Hunter, "Finite Element Analysis of Cardiac Muscle Mechanics," Ph.D. thesis, University of Oxford, Oxford, UK, 1975.
11. D. Noble and Y. Rudy, "Models of Cardiac Ventricular Action Potentials: Iterative Interaction Between Experiment and Simulation," *Phil. Trans. Roy. Soc. Lond. A* **359**, No. 1783, 1127–1142 (2001).

12. A. G. Kléber and Y. Rudy, "Basic Mechanisms of Cardiac Impulse Propagation and Associated Arrhythmias," *Physiol. Rev.* **84**, No. 2, 431–488 (2004).
13. F. F. T. Ch'en, R. D. Vaughan-Jones, K. Clark, and D. Noble, "Modelling Myocardial Ischaemia and Reperfusion," *Prog. Biophys. Mol. Biol.* **69**, No. 2–3, 515–538 (1998).
14. P. J. Mulquiney, N. P. Smith, K. Clark, and P. J. Hunter, "Mathematical Modelling of the Ischaemic Heart," *Nonlinear Anal.* **47**, No. 1, 235–244 (2001).
15. J. J. Rice, M. S. Jafri, and R. L. Winslow, "Modeling Short-Term Interval-Force Relations in Cardiac Muscle," *Amer. J. Physiol. Heart Circ. Physiol.* **278**, No. 3, H913–H931 (2000).
16. D. P. Nickerson, N. P. Smith, and P. J. Hunter, "A Model of Cardiac Cellular Electromechanics," *Phil. Trans. Roy. Soc. Lond. A* **359**, No. 1783, 1159–1172 (2001).
17. R. FitzHugh, "Impulses and Physiological States in Theoretical Models of Nerve Membrane," *Biophys. J.* **1**, 445–466 (1961).
18. J. Nagumo, S. Animoto, and S. Yoshizawa, "An Active Pulse Transmission Line Simulating Nerve Axon," *Proc. Inst. Radio Eng.* **50**, 2061–2070 (1962).
19. A. L. Hodgkin and A. F. Huxley, "A Quantitative Description of Membrane Current and Its Application to Conductance and Excitation in Nerve," *J. Physiol.* **117**, No. 4, 500–544 (1952).
20. J. M. Rogers and A. McCulloch, "A Collocation–Galerkin Finite Element Model of Cardiac Action Potential Propagation," *IEEE Trans. Biomed. Eng.* **41**, No. 8, 743–757 (1994).
21. R. R. Aliev and A. V. Panfilov, "A Simple Two-Variable Model of Cardiac Excitation," *Chaos Solitons Fractals* **7**, No. 3, 293–301 (1996).
22. C.-H. Luo and Y. Rudy, "A Model of the Ventricular Cardiac Action Potential. Depolarisation, Repolarisation, and Their Interaction," *Circ. Res.* **68**, No. 6, 1501–1526 (1991).
23. C.-H. Luo and Y. Rudy, "A Dynamic Model of the Cardiac Ventricular Action Potential. I. Simulations of Ionic Currents and Concentration Changes," *Circ. Res.* **74**, No. 6, 1071–1096 (1994).
24. D. Noble, A. Varghese, P. Kohl, and P. Noble, "Improved Guinea-Pig Ventricular Cell Model Incorporating a Diadic Space, i_{Kr} and i_{Ks}, Length- and Tension-Dependent Processes," *Can. J. Cardiol.* **14**, No. 1, 123–134 (1998).
25. C. E. Clancy and Y. Rudy, "Linking a Genetic Defect to Its Cellular Phenotype in a Cardiac Arrhythmia," *Nature* **400**, No. 6744, 566–569 (1999).
26. C. E. Clancy and Y. Rudy, "Na$^+$ Channel Mutation that Causes Both Brugada and Long-QT Syndrome Phenotypes: A Simulation Study of Mechanism," *Circulation* **105**, No. 10, 1208–1213 (2002).
27. P. Kohl, K. Day, and D. Noble, "Cellular Mechanisms of Cardiac Mechano-Electric Feedback in a Mathematical Model," *Can. J. Cardiol.* **14**, No. 1, 111–119 (1998).
28. N. P. Smith, D. P. Nickerson, E. J. Crampin, and P. J. Hunter, "Multiscale Computational Modelling of the Heart," *Acta Numer.* **13**, 371–431 (2004).
29. R. Plonsey and R. C. Barr, "Current Flow Patterns in Two-Dimensional Anisotropic Bisyncytia with Normal and Extreme Conductivities," *Biophys. J.* **45**, No. 3, 557–571 (1984).
30. C. S. Henriquez, "Simulating the Electrical Behaviour of Cardiac Tissue Using the Bidomain Model," *Crit. Rev. Biomed. Eng.* **21**, No. 1, 1–77 (1993).
31. K. Tomlinson, A. J. Pullan, and P. J. Hunter, "A Finite Element Method for an Eikonal Equation Model of Myocardial Excitation Wavefront Propagation," *SIAM J. Appl. Math.* **63**, No. 1, 324–350 (2002).
32. P. J. Hunter, P. A. McNaughton, and D. Noble, "Analytical Models of Propagation in Excitable Cells," *Prog. Biophys. Mol. Biol.* **30**, No. 2/3, 99–144 (1975).
33. D. Hooks, K. Tomlinson, S. G. Marsden, I. LeGrice, B. H. Smaill, A. J. Pullan, and P. J. Hunter, "Cardiac Microstructure: Implications for Electrical Propagation and Defibrillation in the Heart," *Circ. Res.* **91**, No. 4, 331–338 (2002).
34. P. J. Hunter, A. J. Pullan, and B. H. Smaill, "Modeling Total Heart Function," *Annu. Rev. Biomed. Eng.* **5**, 147–177 (2003).
35. P. J. Hunter, M. P. Nash, and G. B. Sands, "Computational Electromechanics of the Heart," in *Computational Biology of the Heart*, A. V. Panfilov and A. V. Holden, Eds., John Wiley, New York, 1997, pp. 345–407.
36. M. P. Nash and A. V. Panfilov, "Electromechanical Model of Excitable Tissue to Study Reentrant Cardiac Arrhythmias," *Prog. Biophys. Mol. Biol.* **85**, No. 2–3, 501–522 (2004).
37. A. V. Panfilov, R. H. Keldermann, and M. P. Nash, "Self-Organized Pacemakers in a Coupled Reaction–Diffusion–Mechanics System," *Phys. Rev. Lett.* **95**, No. 25, 258104 (2005).
38. T. P. Usyk, I. J. LeGrice, and A. D. McCulloch, "Computational Model of Three-Dimensional Cardiac Electromechanics," *Comput. Visual Sci.* **4**, No. 4, 249–257 (2002).
39. R. C. P. Kerckhoffs, P. H. M. Bovendeerd, J. C. S. Kotte, F. W. Prinzen, K. Smits, and T. Arts, "Homogeneity of Cardiac Contraction Despite Physiological Asynchrony of Depolarization: A Model Study," *Ann. Biomed. Eng.* **31**, No. 5, 536–547 (2003).
40. R. C. P. Kerckhoffs, O. P. Faris, P. H. M. Bovendeerd, F. W. Prinzen, K. Smits, E. R. McVeigh, and T. Arts, "Timing of Depolarization and Contraction in the Paced Canine Left Ventricle: Model and Experiment," *J. Cardiovasc. Electrophysiol.* **14**, No. 10 (Suppl.), S188–S195 (2003).
41. D. P. Nickerson, N. P. Smith, and P. J. Hunter, "New Developments in a Strongly Coupled Cardiac Electromechanical Model," *Europace* **7**, Suppl. 2, s118–s127 (2005).
42. F. Fenton and A. Karma, "Vortex Dynamics in Three-Dimensional Continuous Myocardium with Fiber Rotation: Filament Instability and Fibrillation," *Chaos* **8**, No. 1, 20–47 (1998).
43. P. J. Hunter, A. McCulloch, and H. E. D. J. ter Keurs, "Modelling the Mechanical Properties of Cardiac Muscle," *Prog. Biophys. Molec. Biol.* **69**, No. 2–3, 289–331 (1998).
44. C. E. Clancy and Y. Rudy, "Cellular Consequences of HERG Mutations in the Long QT Syndrome: Precursors to Sudden Cardiac Death," *Cardiovasc. Res.* **50**, No. 2, 301–313 (2001).
45. M. S. Jafri, J. J. Rice, and R. L. Winslow, "Cardiac Ca^{2+} Dynamics: The Role of Ryanodine Receptor Adaptation and Sarcoplasmic Reticulum Load," *Biophys. J.* **74**, No. 3, 1149–1168 (1998).
46. S. A. Niederer, P. J. Hunter, and N. P. Smith, "A Quantitative Analysis of Cardiac Myocyte Relaxation: A Simulation Study," *Biophys. J.* **90**, No. 5, 1697–1722 (2006).
47. M. P. Nash and P. J. Hunter, "Computational Mechanics of the Heart," *J. Elasticity* **61**, No. 1–3, 112–141 (2000).
48. P. Nielsen, I. LeGrice, B. H. Smaill, and P. J. Hunter, "Mathematical Model of Geometry and Fibrous Structure of the Heart," *Amer. J. Physiol. Heart Circ. Physiol.* **260**, No. 29, H1365–H1378 (1991).
49. C. Stevens and P. J. Hunter, "Sarcomere Length Changes in a Model of the Pig Heart," *Prog. Biophys. Mol. Biol.* **82**, No. 1–3, 229–241 (2003).
50. J. W. Fernandez and P. J. Hunter, "An Anatomically Based Finite Element Model of Patella Articulation: Towards a Diagnostic Tool," *Biomechan. Model. Mechanobiol.* **4**, No. 1, 20–38 (2005).
51. J. W. Fernandez, M. L. Buist, D. P. Nickerson, and P. J. Hunter, "Modelling the Passive and Nerve Activated Response of the Rectus Femoris Muscle to a Flexion Loading: A Finite Element Framework," *Med. Eng. Phys.* **27**, No. 10, 862–870 (2005).
52. S. Muttaiyah, M. Tawhai, and W. Thorpe, "Modelling an Active Bronchial Airway," Part IV, Biomedical Engineering Project, Bioengineering Institute, The University of Auckland, Auckland, New Zealand, 2004.

Received September 27, 2005; accepted for publication December 20, 2005; Internet publication June 27, 2006

David Nickerson *Bioengineering Institute, The University of Auckland, Auckland, New Zealand (d.nickerson@auckland.ac.nz).* Dr. Nickerson received his Ph.D. degree in bioengineering at The University of Auckland in 2005. His work focused on computational modeling of cardiac electromechanics and the use of XML languages to specify simulation-specific mathematical models. He currently works as a postdoctoral research fellow in the Bioengineering Institute at The University of Auckland, where he is developing anatomically and biophysically based models of cardiac electromechanics and the computational tools required to solve these models as part of a Wellcome Trust (UK)-funded Heart Physiome project. Dr. Nickerson also continues to play an active role in the development of the CellML language and its associated software development.

Martyn Nash *Bioengineering Institute, The University of Auckland, Auckland, New Zealand (martyn.nash@auckland.ac.nz).* Dr. Nash is a Research Scientist at the Bioengineering Institute and a Senior Lecturer in Engineering Science at The University of Auckland, New Zealand. He received his B.E. degree with first-class honors in engineering science in 1991, and his Ph.D. degree, focusing on finite element modeling of ventricular mechanics, in 1998, both from The University of Auckland. From 1997 to 2002, he worked as a postdoctoral research scientist in the Laboratory of Physiology at Oxford University, focusing on the characterization of the electrical activity of animal and human hearts under normal and pathological conditions. Since 2003, Dr. Nash has been engaged in undergraduate teaching of the Biomedical Engineering degree program at The University of Auckland. His primary research interests are concerned with understanding the electrical and mechanical function of the heart, with particular emphasis on elucidating mechanisms of arrhythmia and fibrillation.

Poul Nielsen *Bioengineering Institute, The University of Auckland, Auckland, New Zealand (p.nielsen@auckland.ac.nz).* Dr. Nielsen received a B.Sc. (physics and mathematics) degree in 1978, a B.E. (engineering science) degree in 1981, and a Ph.D. (finite element description of the architecture of the heart) degree at The University of Auckland in 1987. He subsequently spent 30 months as a postdoctoral fellow at the Biomedical Engineering Unit, McGill University, Montréal, Québec, Canada. He is currently a Research Scientist at the Bioengineering Institute, Senior Lecturer in Engineering Science, and coordinator of the Biomedical Engineering program at The University of Auckland. Dr. Nielsen's research interests include the development of modeling tools and instrumentation associated with soft-tissue mechanics (skin, breast, and brain) and muscle thermodynamics, the creation of XML-based markup languages (CellML and FieldML) to facilitate the exchange of biological models, and the development of ontology and graphically based tools for creating and editing biological models.

Nicolas Smith *Bioengineering Institute, The University of Auckland, Auckland, New Zealand (np.smith@auckland.ac.nz).* Dr. Smith completed an engineering degree in 1993 in the Department of Engineering Science at The University of Auckland. After three years working in industry, he returned to graduate study at The University of Auckland, completing a Ph.D. degree in 1999 in bioengineering, focusing on the development of a mathematical model of coronary blood. He then completed a two-year postdoctoral fellowship in physiology at the University of Oxford. He is currently a Senior Lecturer in the Department of Engineering Science and the leader of the Metabolic Modeling group in the Bioengineering Institute at The University of Auckland. His research interests are focused on the mathematical modeling of metabolism at multiple spatial and temporal scales. This includes coupling of cellular models of contraction and electrophysiology to tissue-scale finite-element models of mechanics and perfusion. With these techniques, a biophysically based framework to elucidate the mechanisms underlying pathologies such as ischemic heart failure is being developed.

Peter Hunter *Bioengineering Institute, The University of Auckland, Auckland, New Zealand (p.hunter@auckland.ac.nz).* Dr. Hunter completed an engineering degree in 1971 in theoretical and applied mechanics at The University of Auckland, New Zealand, a Master of Engineering degree in 1972, also at The University of Auckland, for solving the equations of arterial blood flow, and a D.Phil. (Ph.D.) degree in physiology at the University of Oxford in 1975 for finite-element modeling of ventricular mechanics. His major research interests since then have been modeling many aspects of the human body using specially developed computational algorithms and an anatomically and biophysically based approach that incorporates detailed anatomical and microstructural measurements and material properties into continuum models. The interrelated electrical, mechanical, and biochemical functions of the heart, for example, have been modeled in the first "physiome" model of an organ. As the current Co-chairman of the Physiome Committee of the International Union of Physiological Sciences, Dr. Hunter is helping to lead the international Physiome Project, which aims to use computational methods for understanding the integrated physiological function of the body in terms of the structure and function of tissues, cells, and proteins. He is currently Director of the Bioengineering Institute at The University of Auckland and Director of Computational Physiology at Oxford University.