INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN BIOMEDICAL ENGINEERING
Int. J. Numer. Meth. Biomed. Engng. (2011)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/cnm.1443

A massively parallel computational electrophysiology model of the heart

M. Vázquez[1,*,†], R. Arís[1], G. Houzeaux[1], R. Aubry[1], P. Villar[1], J. Garcia-Barnés[2], D. Gil[2] and F. Carreras[3]

[1]*Department of Computer Applications in Science and Engineering, Barcelona Supercomputing Center–Centro Nacional de Supercomputación, Barcelona, Spain*
[2]*Department of Computer Sciences, Centre de Visió per Computador, Universitat Autónoma de Catalunya, Spain*
[3]*Servei de Cardiologia, Hospital de la Santa Creu i Sant Pau, Barcelona, Spain*

SUMMARY

This paper presents a patient-sensitive simulation strategy capable of using the most efficient way the high-performance computational resources. The proposed strategy directly involves three different players: Computational Mechanics Scientists (CMS), Image Processing Scientists and Cardiologists, each one mastering its own expertise area within the project. This paper describes the general integrative scheme but focusing on the CMS side presents a massively parallel implementation of computational electrophysiology applied to cardiac tissue simulation. The paper covers different angles of the computational problem: equations, numerical issues, the algorithm and parallel implementation. The proposed methodology is illustrated with numerical simulations testing all the different possibilities, ranging from small domains up to very large ones. A key issue is the almost ideal scalability not only for large and complex problems but also for medium-size meshes. The explicit formulation is particularly well suited for solving this highly transient problems, with very short time-scale. Copyright © 2011 John Wiley & Sons, Ltd.

Received 7 September 2010; Revised 1 February 2011; Accepted 27 February 2011

KEY WORDS: computational electrophysiology; parallelization; finite element methods

1. INTRODUCTION

In [1], a statement of Daniel D. Streeter Jr. from [2] is reproduced: 'No man-made structure is designed like a heart. Considering the highly sophisticated engineering evidenced in the heart, it is not surprising that our understanding of it comes so slowly'. This paper presents the development of high-performance computational mechanics (HPCM) tools to help researchers and cardiologists to get a deeper insight the problem. The need for efficient HPCM tools, capable of running at top speed in currently available computational resources is also reflected in the paper of Bordas *et al.* [3]. The present work aims at filling this gap.

Heart simulation is a complex problem. Hunter *et al.* [4] provide an authoritative review that describes the International Union of Physiological Sciences (IUPS) Physiome Project. First of all, the issue is extremely multi-disciplinary, with various communication difficulties among the participants. Next, many scales and different physics are strongly coupled. As stated by LeGrice *et al.* [5] the simulation scales covers eight orders of magnitude, 'from the geometry of the cardiac chambers down to descriptions of ion channel or T-tubule density and distribution' which are tightly

*Correspondence to: M. Vázquez, Department of Computer Applications in Science and Engineering, Barcelona Supercomputing Center–Centro Nacional de Supercomputación, Barcelona, Spain.
†E-mail: mariano.vazquez@bsc.es

linked. Owing to the complexity of the problem, large-scale computational resources are needed, rendering the parallelization issue a key point. Finally, experimental validation is inaccurate and difficult to obtain, when compared to more traditional engineering problems on cars, planes or pumps. This paper presents the multi-disciplinary work, focusing on the HPCM aspects. The goal of this project is to apply the simulation tool to study the three-dimensional electrical propagation in patient-specific ventricular geometries in the fastest way to give cardiologists an additional source of information to understand how the heart works. Besides, from its very conception, the simulation tool is flexible enough to introduce all the necessary modelization improvements preserving the main features of the computational tool, particularly the scalability.

The paper is organized as follows. Section 2 describes briefly the problem we are dealing with, including the simulation model. Next, Section 3 is described through the governing differential equations and their backing set of hypothesis. Section 4 focuses on the numerical methods, including improvements in the numerical treatment of the ionic current terms, shock-capturing techniques and treatment of strong fiber gradients. The most important computational aspects, including parallelization, are described in Section 5. All that is proposed in the paper is presented through a set of numerical examples. The paper ends with the discussion, conclusions and future lines.

2. PROBLEM STATEMENT: THE WORKING FRAMEWORK

The working framework can be divided into two parts. First, how to establish the input for the simulation and second, the simulation model itself. The first problem consists in developing the tools for extracting the heart geometry from medical images that are processed to produce patient-sensitive geometries. For each patient, the functionality and the geometry of the ventricles are obtained using magnetic resonances imaging (MRI) techniques. In addition, some studies made in *ex vivo* hearts give the average model of the fibers. Notice that special treatment must be designed for the outcomes of these experiments in order to be used as inputs for a computer simulation. The next problem is to devise the simulation model. A computational mesh is generated using both geometry and fibers at the best resolution possible. After that, hundreds of processors are used to run the simulation in some minutes. Finally, the results are post-processed. This paper focuses on the simulation part. In the conference papers [6, 7], the image processing issues and the simulation are briefly described.

2.1. The simulation model

During the past two decades, different modeling approaches have been developed for bridging the physical scales corresponding to the different levels of the heart structural description in an efficient and a realistic way [8, 9]. This process is part of a general one called integrative biological modeling, which is an active research topic that includes structural, functional and data integrations [10]. Currently, the researchers mainstream for bridging scales correspond both to cellular and reaction–diffusion systems. The former is based on two components: discrete network of cells representing the spatial structure and communication rules between cells in order to reproduce the propagation wave. The latter is based on conservation laws and differential equations to describe the excitation and propagation process in cardiac tissue. The present paper is based on the second line of action, mainly relying in the models studied in works like [11, 12]. The computational framework used is The Alya System for large-scale computational mechanics, which is developed and maintained in the CASE Department of the Barcelona Supercomputing Center (see, for instance, [13–16]). The finite element space discretization and the parallelization based on automatic domain decomposition for distributed memory facilities are the main features of this strategy. Fully implicit and explicit schemes are implemented. The resulting code is able to run using elements of different space order (P1, P2, Q1, Q2) and higher order time schemes, in bi and tri-dimensional domains of a non-homogeneous anisotropic excitable media. Fully implicit and explicit schemes are implemented (see also [17]).

DOI: 10.1002/cnm

3. PHYSIOLOGICAL MODELS

Well-defined mathematical and numerical models of cellular electrophysiology and action potential propagation are of great importance for researching and understanding different electrophysiological phenomena [18, 19]. Nowadays, a challenge is to design tools for using computational models in medical applications, such as realistic simulations for prevention, diagnosis and therapy of cardiac diseases [20, 21]. From the early 60s, generic models of single-cell (microscopic level) have been created for a wide range of species and cardiac cell types, see an extensive list in [22]. These models have achieved a high degree of detail in the description of cellular ionic currents and dynamics of several intracellular structures. They have been used to advance the understanding of processes influencing specific patho-physiological responses, and have also been successful in describing cell properties. Despite all these advantages, it is impracticable to derive a whole heart (macroscopic level) by modeling every single cell that it contains [10]. The most obvious reason of this impossibility is the huge computing requirements, which make the problem computationally intractable.

Since the mid-forties, several methods have been developed in order to simulate the electrical propagation in the excitable media (macroscopic or multicellular level) such as the cardiac tissue: reaction–diffusion systems, cellular automata and hybrid models (again, [22]). Initially, they were used to reproduce excitation propagation in two-dimensional sheets of cardiac muscle. In the last years, several works have shown that these computational models can also be applied to describe three-dimensional phenomena [11, 12, 23]. These models do not take into account microscopic cellular behavior and, typically, the set of parameters that determines their action have scarce (or null) physiological meaning.

3.1. The governing equations

In this paper, the electrophysiology potential $\phi(x_i, t)$ is modeled using a propagation equation through a macroscopic continuous media combined with a microscopic model, properly projected to the large scales. In this paper, we adopt some of the available physiological propagation models taking for granted that the bridging scales' projections are well done (it is worth to mention that this fact is highly questionable, leaving further analysis for future work).

The basic form of the electrical activation potentials ϕ_α propagation equation is[‡]:

$$\frac{\partial \phi_\alpha}{\partial t} = \frac{\partial}{\partial x_i}\left(D_{ij}\frac{\partial \phi_\alpha}{\partial x_j}\right) + L(\phi_\alpha). \tag{1}$$

The Latin subscripts count the space dimension of the problem. The Greek subscripts count the number of activation potentials involved, $\alpha = 1$ for monodomain models and $\alpha = 1, 2$ for bidomain ones, extracellular and intracellular (like in [24]). Depending on that, some of the terms of the propagation equation present differences. The equation's right-hand side represents the transient macroscopic model, based on the well-known *continuum cable equation* [22]. The diffusion term is governed by the diffusion tensor D_{ij}. The equation is set in a fixed reference frame and D_{ij} must describe the cable (i.e. cardiac tissue fibers) orientation in the fixed reference frame. In this paper, we will show the results for two activation potential propagation models in a monodomain scheme: Fitzhugh–Nagumo (FHN) [25] and Fenton–Karma (FK) [12, 26] with an anisotropic fiber field. In the following section, both models are described.

3.2. Ionic current models: *Fitzhugh–Nagumo and Fenton–Karma*

When considering a monodomain scheme, $\alpha = 1$ and Equation (1) can be written as follows:

$$C_m \frac{\partial \phi}{\partial t} = \frac{\partial}{\partial x_i}\left(\frac{D_{ij}}{S_v}\frac{\partial \phi}{\partial x_j}\right) + I_{\text{ion}}, \tag{2}$$

[‡]Einstein convention on repeated indexes is used.

DOI: 10.1002/cnm

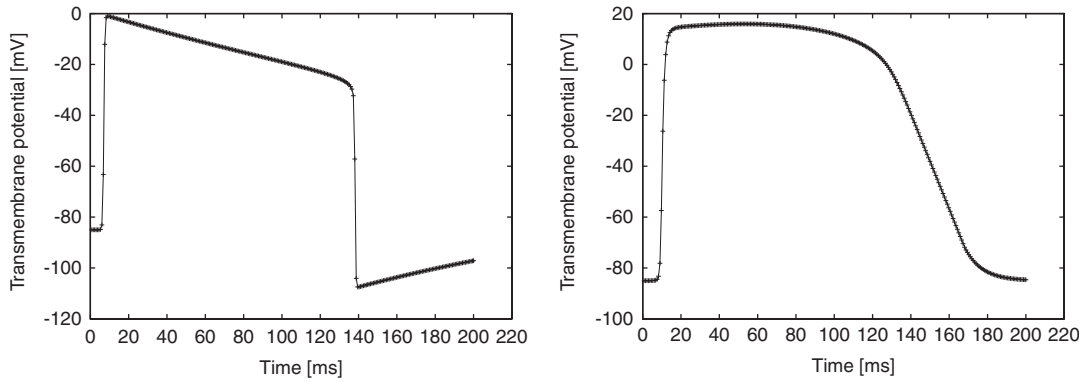

Figure 1. Fitzhugh–Nagumo and Fenton–Karma transmembrane potential ϕ temporal evolution for a given position. The shape differences between them can be reduced by parameter fine-tuning.

where C_m and S_v are constants of the models, the membrane capacitance ($\mu F\,cm^{-2}$) and the surface-to-volume ratio, respectively. The total membrane ionic current is I_{ion} ($\mu A\,cm$). FHN and FK differ in the way they model the ionic current I_{ion}, which is complemented with ordinary differential equations for the so-called gate potentials. In Appendix A, the parameters for both models are listed.

Fitzhugh–Nagumo model: In this model, to the ionic current definition, a gate potential W equation is added:

$$\begin{aligned} I_{ion} &= c_1\phi(\phi-c_3)(\phi-1)+c_2W, \\ \frac{\partial W}{\partial t} &= \varepsilon(\phi-\gamma W). \end{aligned} \tag{3}$$

W is called the recovery potential. Constants c_1, c_2 and c_3 define the shape of the propagation wave (Figure 1) and ε and γ control the recovery potential evolution.

Fenton–Karma models: FK models are of higher complexity, as they decompose the ionic current into three, each part with its own behavior. Additionally, it solves two additional ODE's for the so-called inactivation–deactivation gates

$$\begin{aligned} I_{ion} &= J_{fi}(\phi,V)+J_{so}(\phi)+J_{fi}(\phi,W), \\ \frac{\partial W}{\partial t} &= \Theta(\phi_c-\phi)\frac{(1-W)}{\tau_W^-}-\Theta(\phi-\phi_c)\frac{W}{\tau_W^+}, \\ \frac{\partial V}{\partial t} &= \Theta(\phi_c-\phi)\frac{(1-V)}{\tau_V^-}-\Theta(\phi-\phi_c)\frac{V}{\tau_V^+}, \\ J_{fi}(\phi,V) &= -\frac{V}{\tau_d}\Theta(\phi-\phi_c)(1-\phi)(\phi-\phi_c), \\ J_{so}(\phi) &= \frac{\phi}{\tau_0}\Theta(\phi_c-\phi)+\frac{1}{\tau_r}\Theta(\phi-\phi_c), \\ J_{si}(\phi,W) &= -\frac{W}{2\tau_{si}}(1+\tanh[k(\phi-\phi_c^{si})]) \end{aligned} \tag{4}$$

J_{fi} is the fast inward current, which is related to the depolarization of the membrane, being analogous to the sodium current in the Beeler–Reuter and Luo–Rudy models. J_{so} is the slow outward current, which deals with the repolarization of the membrane, analogous to the potassium current. J_{si} is a slow inward current analogous to the calcium current. Θ is the Heaviside function. Depending on the choice of the model parameters, the FK mimics the Beeler–Reuter, modified Beeler–Reuter, modified Luo–Rudy I or a Guinea Pig model (see [26]).

DOI: 10.1002/cnm

4. NUMERICAL METHODS

This section describes the numerical method used to discretize the aforementioned equations. Discretization is carried out using a variational formulation together with an FEM-based spacial discretization and an FD-based Euler time integration. A similar scheme was introduced in [17]. The weak form of Equation (2) is: find $\phi \in \mathscr{V}$ such that $\forall \psi \in \mathscr{W}$ which verifies

$$\int_\Omega \psi \frac{\partial \phi}{\partial t} \, \mathrm{d}V = \int_{\partial\Omega} \psi \frac{D_{ij}}{C_m S_v} \frac{\partial \phi}{\partial x_j} n_j \, \mathrm{d}S - \int_\Omega \frac{D_{ij}}{C_m S_v} \frac{\partial \psi}{\partial x_i} \frac{\partial \phi}{\partial x_j} \, \mathrm{d}V + \int_\Omega \psi I_{\mathrm{ion}} \, \mathrm{d}V. \tag{5}$$

Spaces $\mathscr{W}$ and $\mathscr{V}$ are the usual FEM trial and interpolation function spaces (see, for instance, [27]), defined over space tessellations using time-independent compact support functions. The simulation domain is Ω and its boundary is $\partial\Omega$, both of them fixed in time. The first term on the right-hand side is used to impose Neumann boundary conditions, for instance when considering the effect of the media surrounding the simulation domain. As a first approximation, we will set the fluxes on the boundaries to zero. No Dirichlet conditions are directly imposed. The time evolution of the activation potential is triggered by initial localized impulses in some predefined locations.

The weak form equation (5) is discretized in space using the finite element method and, in time, with a finite differences trapezoidal rule allowing backwards or forward Euler and Crank–Nicholson formulations. Then, the diffusion matrix sets if the time integration scheme is implicit or explicit. Both schemes have been programmed in Alya, but due to the highly transient character of the propagating activation potential wave (1), to the large amount of degrees of freedom (2) and to the strong non-linearities of the ionic current models (3), we have observed that the explicit scheme is more convenient to solve large problems and the implicit one is better when the problems are small. In [17], we have introduced a linearization treatment which has been decisive in implicit schemes with larger time steps, solving small problems. In this paper, we will focus on large-scale problems solved explicitly, which due to the shorter time steps do not need a special linearization treatment. A last comment on parallel implicit methods for electrophysiology: due to space anisotropy coming from fibers, the diffusion tensor is non-symmetric; therefore, precluding the use of Conjugate Gradient methods. This fact favors the use of methods such as the one introduced in [14], an efficient parallel strategy implemented in Alya for non-symmetric problems.

Bidomain and monodomain models: The bidomain models require the additional solution of a Poisson equation for the extracellular activation potential. In this paper, bidomain models are left aside. However, the resulting additional equation can be very efficiently solved with a parallelized deflated conjugated gradient (DCG) already implemented in Alya and described in detail in [28].

Numerical integration of the I_{ion} term: The FEM-based formulation leads to a mass matrix in the time derivative term, a diffusion matrix with non-isotropic diffusion tensor and a non-linear ODE-like term for the ionic current. In the explicit case, only the time derivative term is implicitly solved with a so-called mass matrix that must be inverted and a residual nodal vector explicitly computed at each time step. One alternative to yield a diagonal mass matrix that can be trivially inverted is lumping, which consists of a special treatment of the rows of the consistent mass matrix (non-diagonal) to diagonalize it. Another alternative is to take profit of the FEM properties using a different integration scheme to compute the mass matrix. Figure 2 (left) shows three integration cases for triangles: two open rules with either one or three integration points and one closed rule with three integration points. 'Closed' means that integration points are coincident with nodes. When the closed rule is used, nodal locality is preserved because shape functions are built by setting them to one in its own node and zero in the rest of them. From Figures 2–5 this effect can be assessed. Figure 2 describes the initial condition of an activation potential propagation through anisotropic medium ($k_x = 3k_y$), using the FHN model. Figure 3 shows a sequence of the propagation obtained when the closed rule (top) and the open rule (bottom) is used to integrate the ODE-like local non-linear terms. Note that the use of the open rule for the local term introduces a significant error in speed and propagation behavior. Figure 5 shows the difference in speed propagation by recording the solution at a point near the initial condition. The open rule (dotted curve, so-called 'no lumped') produces a propagation that arrives earlier than the closed rule

DOI: 10.1002/cnm

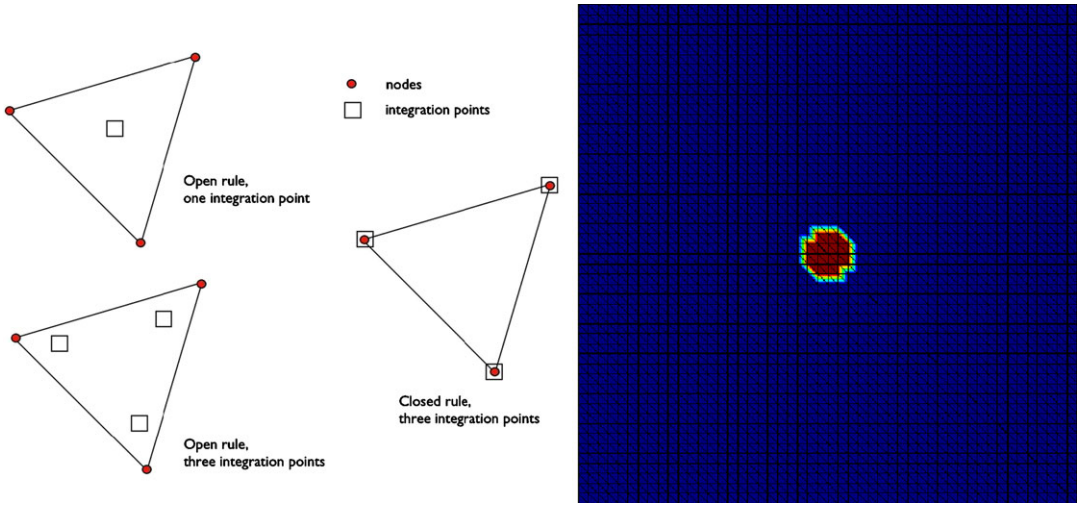

Figure 2. Integration rules (left) and initial condition for the anisotropic 2D potential propagation (right).

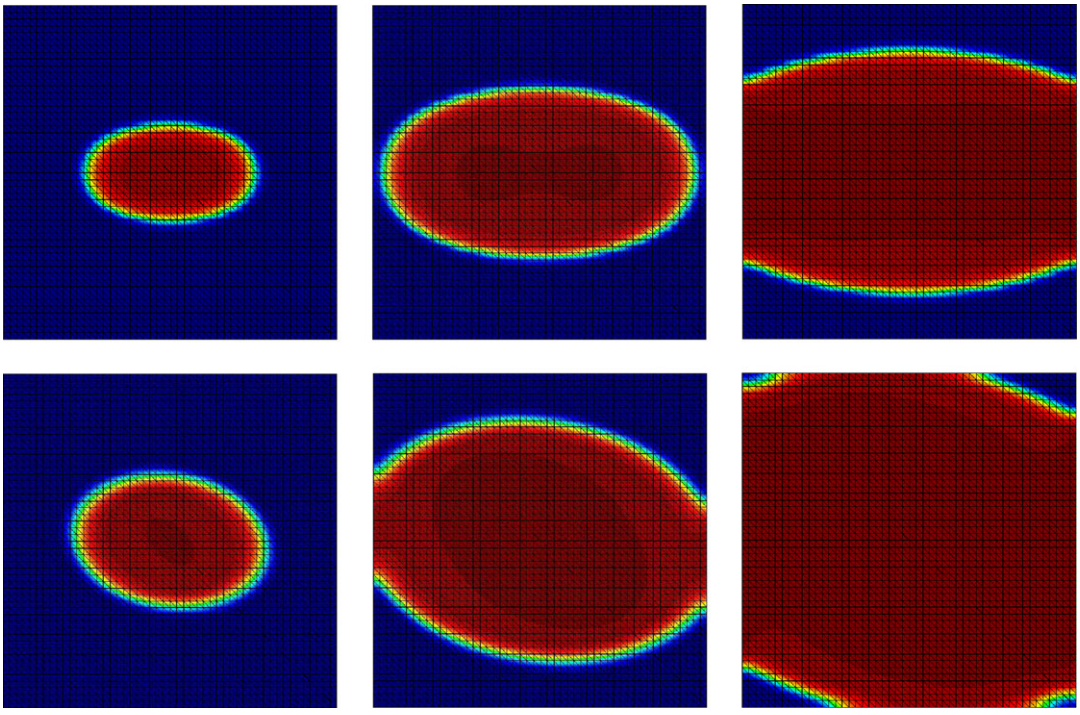

Figure 3. Anisotropic 2D potential propagation sequences: closed integration rule for the ODE-like term (top) versus open integration rule for the ODE-like term (bottom). The first option gives the proper shape and propagation speed.

(so-called 'lumped') to the witness point, as if an additional spurious diffusion is present. Moreover, Figure 4 shows what happens when diffusion is set to zero and the ODE-like is integrated using the open rule, therefore, 'spacializing' its effect. On the other hand, when the closed rule is used, the initial condition, as shown in Figure 2, do not propagate.

The closed rule is also specially well suited for the ODE-like term I_{ion}, which is a local non-linear function of the unknown.

DOI: 10.1002/cnm

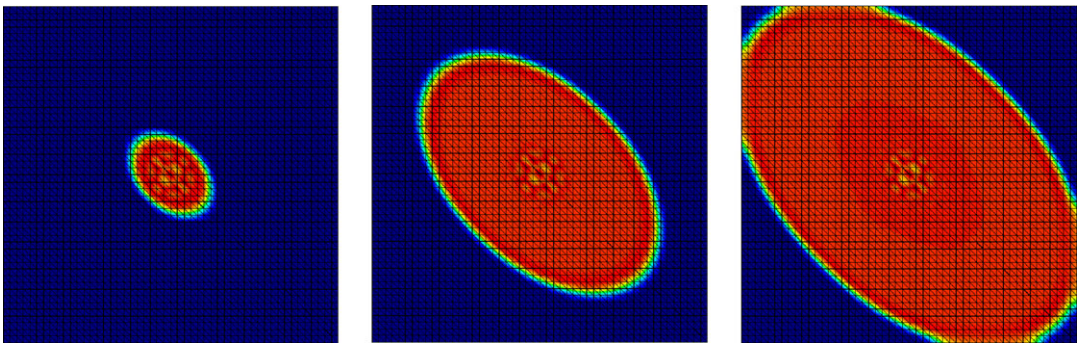

Figure 4. Anisotropic 2D potential propagation sequences: when diffusion is set to zero, by 'spacializing' the ODE-like terms using an open rule, an artificial numerical diffusion is introduced, producing a wrong propagating behavior.

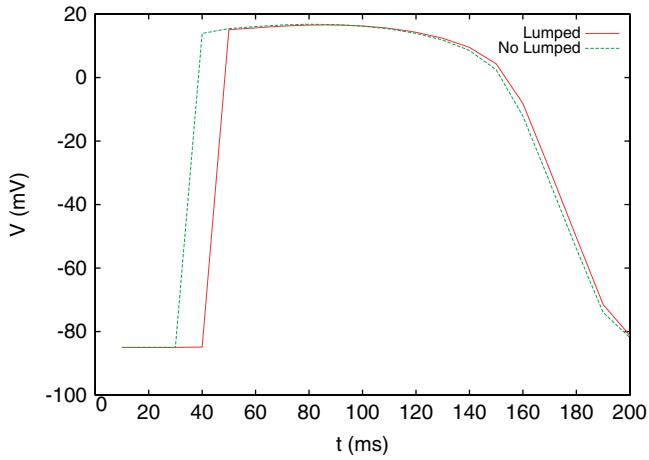

Figure 5. Anisotropic 2D potential propagation: speed differences for the closed rule ('lumped') and the open rule ('no lumped') used in the ODE-like term.

Critical time step: The critical time step for the explicit scheme is computed taken into account *only* diffusion, leaving aside the non-linear term effect, as follows:

$$\Delta t_{\text{crit}} = B\frac{h^2}{2k}, \tag{6}$$

where $k = \max(D_{ii}^{\text{loc}})/C_m S_v$ and B is a safety factor to control the fact that the non-linear term effect on critical time step is neglected. Through numerical experiments, we have observed that Equation (6) is sufficiently conservative for all the cases we tested in the FK and FHN models for P1 and Q1 2- and 3D elements, $B = 1$ being a reliable option. The critical time step is computed for each element and the lowest one is taken as global and used for the explicit time advance. The critical time step is the characteristic time a signal takes to propagate within a discretization unit (an element, a cell, etc.). Signal propagation is due not only to diffusion but also to non-linear terms. However, we have observed that for these problems, the most stringent is that of diffusion. Recall that the critical time step is 'critical' only for explicit integration schemes like the one we are using in this paper. In previous works [17], we have reported to solve the problem using implicit algorithms, which allows larger time steps.

Strong potential gradients treatment in the implicit case: Figure 1 shows how sharp the propagating potential wave depolarization and polarization can be. We have observed that for the implicit cases, where larger time steps are used, apart from the non-linear treatment, a special

DOI: 10.1002/cnm

treatment for large gradients must be included. In [17], we have introduced a shock-capturing method based on [29–31], which was originally devised for convection–diffusion equations and then adapted to fluid flow ones. Shock-capturing techniques are numerical tools that eliminates spurious non-physical oscillations localized around large gradients. These oscillations are typical of non-linear problems and appear when the gradients are not resolved by the spacial discretizations. We propose to cope with this problem using an anisotropic shock-capturing (ASC) technique. Suppose a diffusion–reaction equation like Equation (2), re-written as

$$\frac{\partial \phi}{\partial t} - k_{ij} \frac{\partial^2 \phi}{\partial x_i \partial x_j} + S(\phi) = 0,$$

where we have denoted $S(\phi)$ as the grouping of the non-linear terms and k_{ij} as a generic diffusion tensor. The Einstein summation convention on repeated indexes is used. According to the aforementioned method, a numerical shock-capturing diffusion coefficient k_{sc} is computed elementary, at the Gauss points level:

$$k_{\mathrm{sc}} = \frac{1}{2} \alpha_{\mathrm{sc}} h \frac{|R(V^h)|}{|\nabla V^h|}. \tag{7}$$

This numerical diffusion is properly added in the corresponding (discretized weak form) diffusion terms for each component, comparing with the physical diffusivity of the fibers. In this way, the introduced diffusion is consistent. The supraindexed V^h represents the projection of the continuous action potential on the discretized space. $R(V^h)$ is the space–time (discretized weak form) equation residual, and h is a characteristic length of the discretization element, typically the radius of the inscribed sphere (or circle in 2D). When there are no convection terms (as it is the case), $\alpha_{\mathrm{sc}} = 0.7$, a constant coming from numerical analysis. Note that the numerical diffusion k_{sc} is only present when the discrete equation residual rises. It is worth mentioning that none of the explicit examples used in this paper required shock-capturing techniques, but required only the implicit ones with large time steps.

Space discretization: Alya is endowed with a large data base of element types for the FEM-based space discretization. In this work we show results for simple examples in 2D made of P1 elements (triangles) and P1 and Q1 elements in 3D (tetra and hexahedra). After some tests, tetrahedra have been chosen to solve the large-scale problems. Tetrahedra meshes can be easily adapted to complex geometries and are efficiently generated from raw data, using mesh generators such as Tetgen [32].

Anisotropic diffusion tensor and fiber fields: As discussed in [33–35], the heart-layered structure could strongly influence activation potential propagation (and eventually, mechanical properties). Strong gradient on fiber fields could be used to detect this local sheet-like structure. Diffusion tensor imaging (DTI) [36] is a relatively new imaging modality able to measure diffusion of water molecules, and specially well suited to visualize fibrous tissue such as myocardial fibers. It provides volumetric data containing point-wise real-valued symmetric tensors. These tensors decompose into local eigensystems with the principal eigenvector aligned to the orientation of the fiber. The set of all principal eigenvectors leads to a vector field whose integration (by means of tractographic techniques) provides a unique visualization of the myocardial architecture. Figure 6 shows the tractography of a fiber field for a left ventricle.

As explained above, the diffusion tensor D_{ij} in (1) is written as

$$D_{ij} = C_{ik}^{-1} D_{lk}^{\mathrm{loc}} C_{lj} \tag{8}$$

C_{lk} is the base change matrix from the local fiber-aligned reference frame (a_i, c_i^1, c_i^2) to the global reference frame and D_{lk}^{loc} is the local diagonal diffusion matrix, whose diagonal components are the axial and crosswise fiber diffusion. The crosswise diffusion is equal and typically one-third of the axial diffusion [37, 38]. Although there is *no discontinuity* between the fiber layers, there is a thin region of strong fiber orientation change which can be naturally identified by computing

DOI: 10.1002/cnm

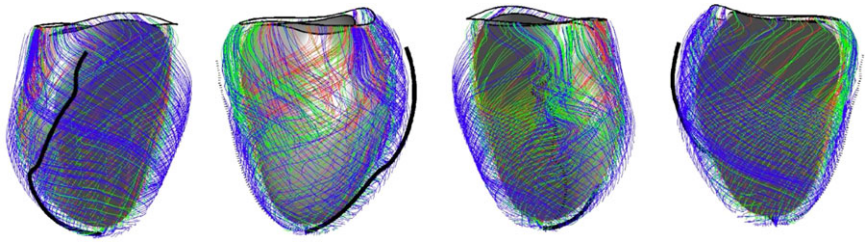

Figure 6. Tractography performed onto the average fiber architecture.

Table I. Anisotropic diffusion, with or without special treatment of large gradients.

Do elementary loop
- Compute the fiber a_i gradient tensor $G_{ij}=\partial a_i/\partial x_j$
- Compute the eigenvalues and eigenvectors of G_{ij}
- If the largest eigenvalue is lower than a threshold
  - Compute c_i^1 and c_i^2 lying in the plane normal to a_i
  - Set an isotropic diffusion in the plane span by c_i^1 and c_i^2
  - Compute $D_{ij}=C_{ik}^{-1}D_{lk}^{\mathrm{loc}}C_{lj}$
- Else, if it is larger
  - Set c_i^1 as the largest eigenvalue direction
  - Compute $c_i^2=\varepsilon_{\mathrm{ijk}}a_j c_k^1$, where $\varepsilon_{\mathrm{ijk}}$ is the Levi-Civita symbol
  - Compute a corrected $c_i^1=\varepsilon_{\mathrm{ijk}}a_j c_k^2$
  - Set an anisotropic diffusion in the plane span by c_i^1 and c_i^2
- End If

End Do

the gradients. Then, for the discretization elements with large fiber gradients, the strongest change direction (i.e. that of the eigenvector with the largest eigenvalue) can be identified locally with the first crosswise direction c_i^1. In general, this c_i^1 is almost normal to the local axial fiber vector a_i. However, to obtain a proper local basis, a small correction is usually required. In brief, the procedure is given in Table I.

5. COMPUTATIONAL ASPECTS: FROM FLEXIBILITY TO PARALLELIZATION

The simulation models presented here are implemented in the Alya System, conceived for simulating coupled multiphysics computational mechanics problems in parallel. Alya [13] is designed from scratch to take profit of parallel architectures with two premises: programming flexibility and parallel efficiency. Programming flexibility means that new Physical models could be rapidly implemented and integrated in the framework. Then, the new models will run either alone or coupled with other models. Additionally, a set of complementary tools will be available to the models to perform specific tasks, such as run in parallel in distributed memory architectures.

A careful design at the early stages in the code development sets the basis for an efficient parallelization, thanks to the code modularity, can be inherited by all the modules running either alone or coupled. Parallelization follows a multi-level strategy, with an outer level based on MPI and automatic mesh partition for distributed memory through interconnected multicores, and one or more inner levels for shared memory within the multicore and/or the accelerators. This strategy is particularly well suited for the typical parallel architectures today and, very likely, in the mid-term future: clusters of multicore processors, eventually complemented with accelerators such as GPUs. It is worth mentioning that this is the currently (2010) most extended architectures, covering from

DOI: 10.1002/cnm

desktops up to large-scale supercomputers. The main advantage of this strategy can be clearly seen with an example. Once the electrophysiological model is up and running in parallel, the mechanical coupling could be attacked in the same code. After some weeks of programming, the coupled model was up and running in parallel (preliminary results in [39, 40]).

In the core of the parallelization outer level lies the automatic mesh partition tool, METIS [41]. It partitions the original mesh into subdomains under two constraints: load-balance and minimal contact surface between subdomains. One example presented in next section shows the action of METIS on unstructured mesh, Figure 14. Mesh partition is done as a pre-process stage following a Master–Slave strategy, previously described in [14].

The Master–Slave strategy: The Master reads the mesh, performs the automatic partition and dumps some output files (for example of the convergence residuals). The slaves build the local element matrices (LHS, left-hand side) and right-hand side (RHS) and are in charge of the resulting system solution in parallel. While running, the Master controls the general situation of the run, checking the stop criteria.

Inter-subdomain communications: In a finite element implementation, only two kinds of communications are necessary between subdomains. The first type of communication consists in exchanging arrays between neighbors with `MPI_Sendrecv` to compute the system matrix (the LHS), for implicit schemes, and the residual (the RHS), for both implicit and explicit schemes. The strategy is the following:

1. For each slave, compute elemental LHS and RHS.
2. For each slave, assemble (scatter) elemental RHS and LHS into global LHS and RHS.
3. Exchange RHS of boundary nodes, the nodes belonging to more than one subdomain, and sum up the contribution.
4. The operations of an iterative solver are matrix–vector multiplications. Then, for each slave, compute matrix–vector multiplication.
5. Exchange the results on the boundary nodes, as was done for the RHS.

The second type of communication is global and of reduce type with `MPI_Reduce`. It is used to compute:

- The critical time step: it is the minimum over the slaves.
- The convergence residual: the sum over all the nodes of the domain. Residuals are required to check the different convergence tolerances of the scheme.
- Scalar products: they take part as iterative solvers.

It is important to stress that we use the mesh partitioning only to *distribute work among the cores*, so the solver gives at every moment the same result running in parallel with many partitions as running sequentially with a unique partition.

Critical issues of the parallelization strategy: In order to have an efficient algorithm to run on thousands of processors, some important aspects of the parallelization must be carefully treated: mesh partitioning, node numbering and communication scheduling.

The *mesh partitioning* is performed using METIS [41]. The main input data of METIS are the element graph and the weight of the vertexes of the graph. In this work, no-hybrid mesh is considered. The difficulty of hybrid meshes for implicit scheme stems from the fact that such an algorithm involves both elemental and nodal loops: elemental for the assembly of the LHS and RHS; nodal in the solver. The element graph is computed by considering as the adjacent elements to element e only the elements sharing a face with e. Figure 7 shows a 1000-partition of the 1M-tetrahedra volume mesh for a left ventricle.

The second important aspect is the *node numbering* of the local (slave) meshes. In order to perform efficient data exchanges between slaves, the local nodes are divided into three categories, as illustrated in Figure 8. The nodes are numbered as follows:

- Interior nodes. They are the nodes that are not shared by another subdomain. They are numbered using the Metis function `METIS_NodeND` [41], which 'computes fill reducing orderings of sparse matrices'.

DOI: 10.1002/cnm

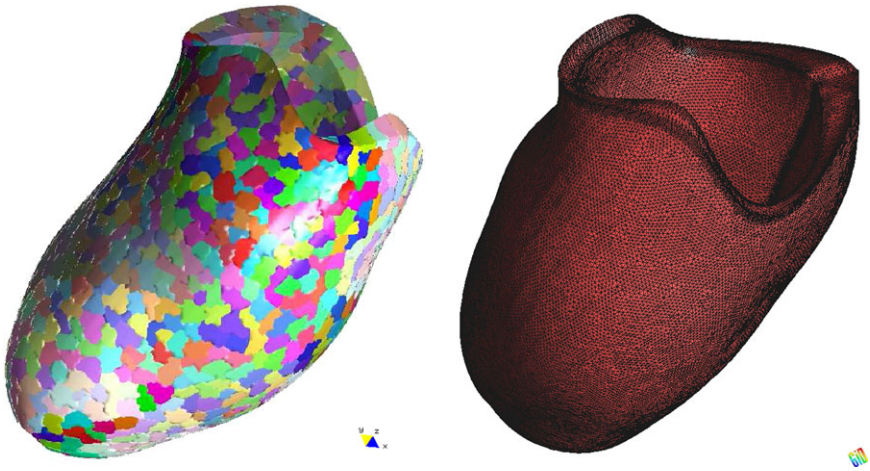

Figure 7. Thousand domains mesh partition performed by Metis (left) on a 1M-tetrahedra unstructured mesh (right) of a single left ventricle.

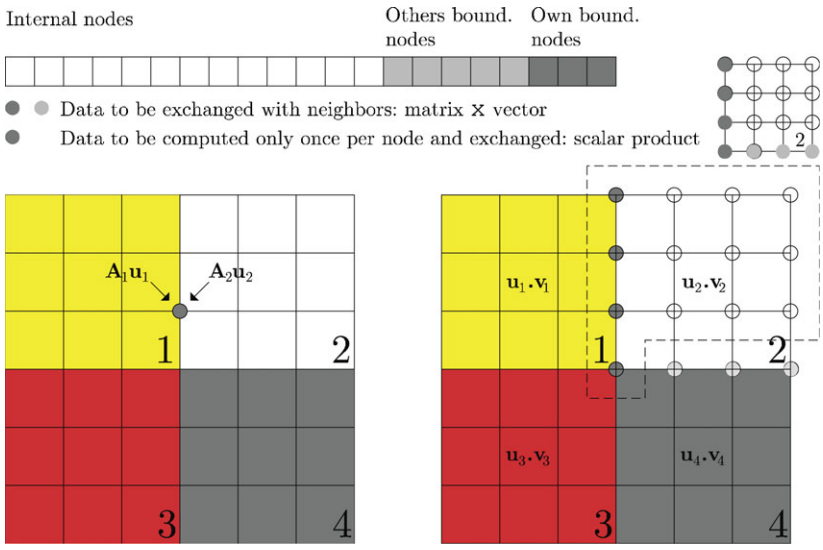

Figure 8. Local node numbering. From light to dark: internal nodes, other subdomains boundary nodes and own boundary nodes.

- Boundary nodes. They are the nodes for which subdomains contribution must be summed up when computing a RHS or a matrix–vector product. The information they carry are repeated for neighboring subdomains. They are divided into Own and Others':
  - Own: Boundary nodes for which the subdomain is responsible. This is useful for computing the convergence residual so that the residuals of the boundary nodes are accounted for only once before performing the sum over the subdomains with `MPI_Reduce`.
  - Others': Boundary nodes belonging to a neighboring subdomain.

The next aspect concerns the *communication scheduling* [42]. Figure 9 illustrates the importance of scheduling on a simple example. For this example, each subdomain has to communicate with all others. On the top part of the figure, we observe the consequence of a bad scheduling, for which the communication is carried out in five steps. For example, during the first step, subdomain 2 cannot communicate with the subdomain 3 until the last one has communicated with subdomain 1.

DOI: 10.1002/cnm

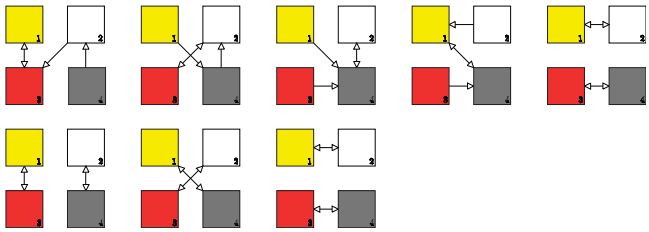

Figure 9. Scheduling strategy on a simple example. (Top) Bad communication scheduling in five steps. (Bottom) Optimum communication scheduling in three steps.

The way to optimize the scheduling is that at each communication step, every subdomain should communicate. The bottom part of the figure illustrates the optimum scheduling for this example, performed in three communication steps.

6. NUMERICAL EXAMPLES

The proposed scheme is assessed with two kinds of numerical examples. A simple academic problem using hexahedra Cartesian domains is used to assess several basic aspects of the algorithm, such as the fiber anisotropy treatment, the sensitivity to errors in the fiber field and the scalability in parallel runs. Next, a large-scale example for two ventricles with fiber distribution is presented.

6.1. Hexahedra domains

Hexahedra domains with different discretization grains are used to adjust some of the algorithm parameters. Figure 10 shows the activation potential as it propagates. The mesh is a structured one, made of 12 000 tetrahedra (2926 nodes). The physical size of the domain is $2\times5\times5$ cm. It is a monodomain model with the FitzHugh–Nagumo scheme, using the models' set of parameters $c_1=10.0$, $c_2=1.0$, $c_3=0.1$, $\varepsilon=0.01$.

6.1.1. Fiber field errors in data acquisition. A second important aspect to assess is the effect of errors in the input fiber field, derived from the data acquisition methodology. Random noise is essentially white and has its origin in thermal Brownian motion of electrons. This noise is superposed in DTI images by two components: noise from the scanner apparatus and noise from the patients body inside the scanner. Additionally, there is the inherent error in considering the DTI output as the fiber field. In order to evaluate the error sensitivity, three different examples of the same slab were run: one with a uniform fiber field, set as the reference, and two more with perturbed fiber fields (10 and 20% of random noise). The percentage of error added to the direction of fibers in the reference is defined by $\alpha\pm\alpha\cdot g\cdot r(x_i)$, where $g=0.1$ or $g=0.2$ and $0\leqslant r(x_i)\leqslant1$ is a random function defined for each point in space x_i. To quantify the error of such a transient problem, we post process the isochrones for each case. The isochrones are contour lines connecting points where the propagating potential reach a marker value at the same time, this marker being the peak value of the activation potential wave. Then, once the polarization wave passes through all the domain, the isochrone field can be compared for the different cases in the discrete L^2 norm, Figure 11.

6.1.2. Strong fiber anisotropy. By identifying strong fiber anisotropy through the gradients principal axes, we propose a natural way of adding sheet-like structure to the tissue. In this way, the fiber crosswise diffusion, which is co-linear with the gradient can take a lower value, 'delaying' the propagation rate when passing through the inter-layer thin region. Figure 12 right shows the activation potential in a slab when crossing from an z-oriented (right half of the slab) to a y-oriented fiber field (left half of the slab). For comparison, on the right side of the figure, the slab with fibers but without special treatment of strong gradients is shown at the same time.

DOI: 10.1002/cnm

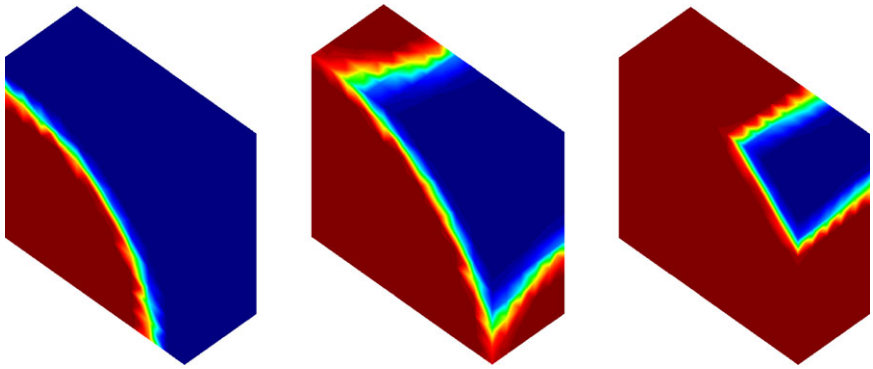

Figure 10. Electrical propagation trough an ideal isotropic cardiac tissue, performed using FitzHugh—Nagumo model. The structured mesh contains 12 000 tetrahedra.

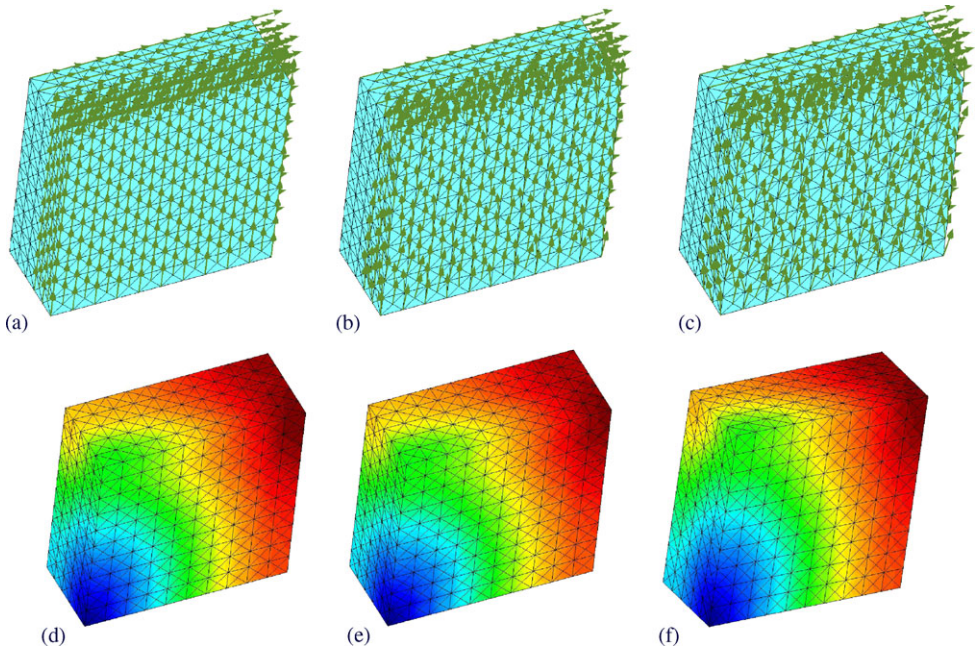

Figure 11. Noise in fiber orientation in 3D. In (a) the distribution of fibers has no noise. In (b) noise in fiber orientation has grown up to 10%. Fibers in (c) has 20% of random noise. The correspondent isochrones are (d)–(f).

6.1.3. Scalability. The slab example, now with a much finer mesh, is used to assess the scalability. Runs with a 10M tetrahedra were carried out, ranging up to 1000 CPUs in BSC's Marenostrum supercomputer. Marenostrum is a distributed memory cluster made of 10 240 IBM PPC processors. The plot in Figure 13 shows the so-called hard scalability, i.e. the scalability measured keeping constant the problem size while increasing the number of processors. The scheme's hard scalability shows an almost linear behavior.

6.2. Two ventricles

This example measures the potential of the proposed methodology for simulating the electrophysiology problem in more realistic cases. We analyze the different steps of the simulation process.

DOI: 10.1002/cnm

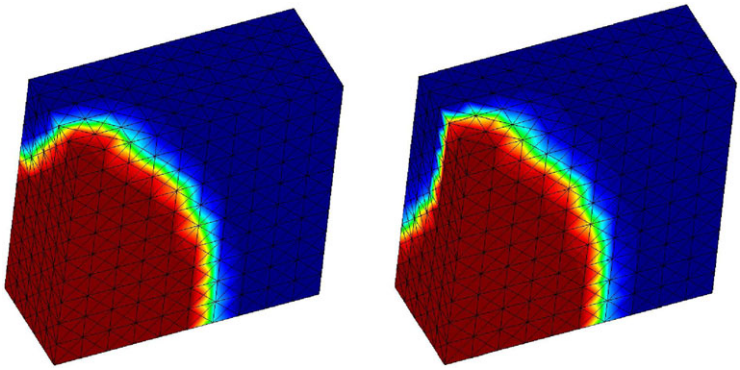

Figure 12. Strong fiber anisotropy. Anisotropic propagation without using the special treatment of large gradients (left) compared to the proposed method (right). Note how the activation propagation is delayed through the region with strong gradients in the fibers.

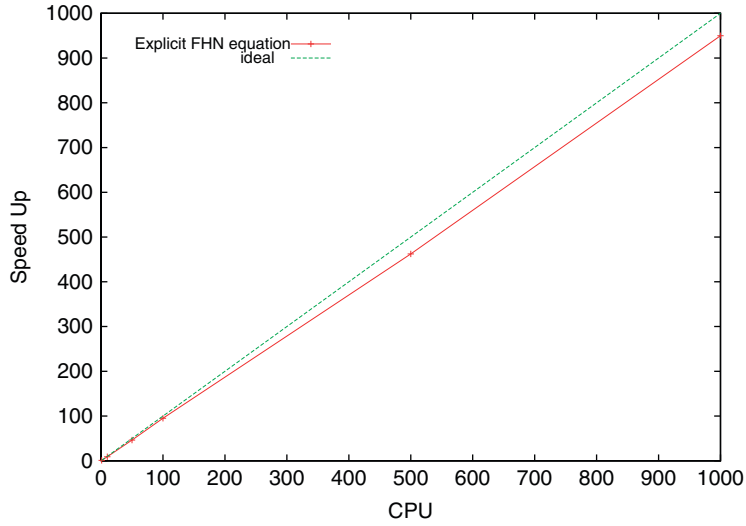

Figure 13. Scalability up to 1000 cores of Marenostrum supercomputer.

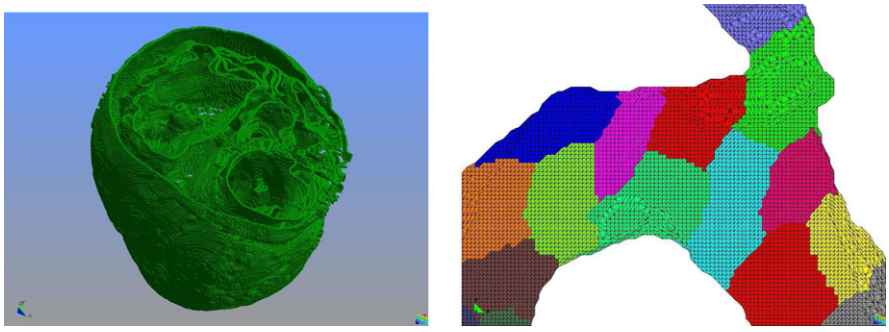

Figure 14. Input data sets (left) and close-up of the tetrahedra mesh produced by Tetgen using the data available at http://www.ccbm.jhu.edu/. Patches (right) are related to the mesh partition carried out by Metis for parallelization purposes.

6.2.1. Data acquisition and medical image processing. DTI data used in this work belong to healthy canine hearts and is freely available at *http://www.ccbm.jhu.edu/*. Apart from fiber architecture, DTI provides, in a single scan, the external anatomy of the whole myocardium by means

DOI: 10.1002/cnm

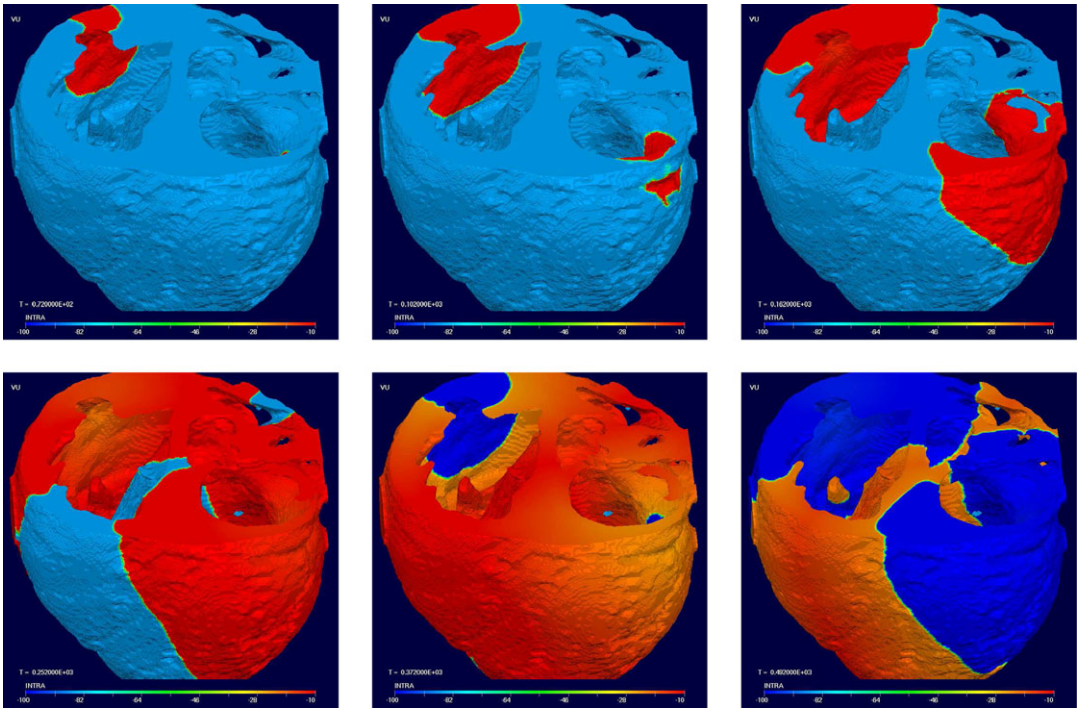

Figure 15. Activation potential propagation.

of the so-called unweighted volumes. The heart geometry is described as a field of points regularly distributed in horizontal cuts of the heart, with a mean resolution of 0.2 mm. Figure 15 (left) shows how data sets are input and the resulting mesh is generated using Tetgen [32]. Together with the point coordinates, the fiber orientation is provided also at each of the points. Tetgen is used next to generate a mesh using the points as nodes.

6.2.2. Simulation. The resulting meshes look like that of Figure 15. In the example shown, the mesh is made of 17M tetrahedra, with a resolution of 0.2–0.15 mm, which is comparable with the resolution of the medical images obtained nowadays. A 2000 ms simulation runs in a bit more than 15 min of wall clock time using 500 cores of Marenostrum, resulting in a mean of 34 K elements per core which comes from the Metis partition. It is worth mentioning that neither queue waiting time nor delays due to hardware failures are not included in wall clock time.

6.2.3. Data analysis. Figure 15 shows the evolution of the activation potential in the heart. The initial condition comes from experimental evidence [43]. Two stimuli spread from the base to the apex with a delay of 30 ms between them. The six pictures are snapshots of the propagation at six different times. Figure 16 shows the isochrones. At the left, those of an isotropic model are shown for reference and comparison with the picture on the right. Note how the isochrones became much tighter packed in the septum and the broken, arrow-like shape in the left ventricle wall, due both cases to the fiber conduction action.

7. DISCUSSION

The purpose of this simulation model is to be capable of reproducing the behavior of the heart with the best degree of accuracy possible in a computer. The resultant simulation tool has to be prepared to include all the Physiological complexity of the currently available models, including data assimilation and parameters fitting. Moreover, as medical imaging progresses, the highest

DOI: 10.1002/cnm

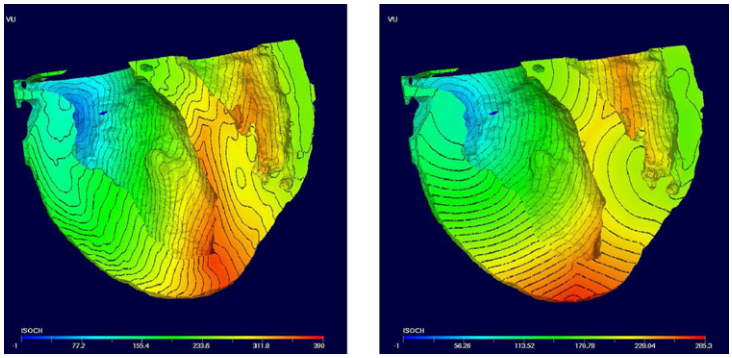

Figure 16. Isochrones.

simulation fidelity is required. For all these reasons, these authors strongly believe that, particularly in this field, an efficient parallel platform is a must. This paper introduces the computational aspects of an Electrophysiology model of the heart, capable of running on thousands of processors and flexible enough to easily build up much more complex models. The 'computational aspects' goes from the numerical algorithms up to the parallel implementation. The paper describes all these points, focusing on the novelties of the scheme.

From the numerical point of view, first we introduce a method to cope with high anisotropic fiber behavior. This method is a zeroth-order strategy to detect sheet-like structures, such as those mentioned in [5]. In this way, these structures could be detected upon the measured data in those parts where fibers change direction abruptly. This fact will certainly influence the electrophysiological behavior and the mechanical deformation, which certainly requires orthotropic models. We have also run some tests adding noise to the fiber field and shown to what extent this noise influences the final results. We have observed that results have a much higher sensitivity to the anisotropic treatment with strong gradients in the fiber field than to the noise in the fiber field. Another novel numerical aspect is how the ODE-like terms in the model are treated. We propose to use a different numerical integration for these terms, similar to that used for the lumped mass matrix. In Figures 2–5, we have shown that if the ODE-like local terms are 'spacialized', the solution presents spurious numerical effects that can lead to plainly wrong results (notably, the case with zero diffusion). These three points are clear examples of how a different numerical treatment, which is basically correct, can yield deep errors. If we are dealing with a single case, some model parameters could be adjusted to somewhat correct the problem. For instance, slightly changing the diffusion to correct speed propagation when no special integration is used in the ODE-like terms. However, it is obvious that this is not the correct strategy when solving many cases, with high dispersion on geometry and parameters, and one wants to assimilate measured data.

On the computational side, we introduce the implementation aspects of the parallel platform Alya. We describe some special parallelization issues, such as mesh partition, communication and data arrangement and storage. The combined flexibility and parallel performance add a great advantage to the proposed approach when extending the model toward data assimilation, multiscales and multiphysics. The present scheme shows almost linear scalability plots for thousands of processors in explicit and implicit formulations. When carefully programmed, this is inherited by all the Physical problems simulated in the platform. Then, in Alya, electrophysiology, mechanical deformation and fluid flow can be coupled retaining this parallel efficiency. On the other hand, we believe that writing separate codes for each problem could represent a big hindrance. We strongly advocate that in order to attain the best parallel efficiency in modern platforms, you should not *parallelize* a sequential code but you must *write* a code sequentially or parallelly.

Ours is not the only and first effort in this direction and for this field. In 2005 Nickerson *et al.* [44], report 1000 ms of simulation of contraction and ejection in a heterogeneous left ventricular model. They run the problem on eight processors in three weeks time using an IBM Regatta P690 high-performance computer. In 2010, researchers in Oxford presented results of scalability

DOI: 10.1002/cnm

from the parallel simulation package Chaste going up to 2048 cores [45–47] and reported 90% scalability up to 512 cores and progressively degraded further. Washio *et al.* [48] presented an algorithm for multi-scale heart simulation that took about 10 h using 6144 cores.

8. CONCLUSIONS AND FUTURE LINES

This paper introduces the computational aspects of an Electrophysiology model of the heart, capable of running efficiently in large-scale parallel facilities and flexible enough to build up a much more complex model. We describe here the physiological models implemented, FHN and Fenton–Karma. From the numerical point of view, we introduce a method of coping with high anisotropic fiber behavior. By proposing the use of a close nodal integration rule for the mass matrix, a consistent local-spatial scheme is obtained. In addition, a shock-capturing technique is proposed, which is required in the implicit scheme because longer time steps increase the non-linear effects. On the computational side, we introduce the implementation aspects of the parallel platform Alya. We describe some special parallelization issues, such as mesh partition, communication and data arrangement and storage. Some numerical examples are used to test the scheme and its implementation. Sensitivity to both fiber field errors and strong gradients treatment is evaluated. Finally, two-ventricle geometry is used to show the simulation platform potential in more realistic cases.

This paper is introductory to a wider Computational Biomechanics strategy. Combining the experience acquired in Electrophysiology and our own background in HPC techniques, we have written a parallel code specific for the computational electrophysiology (CEPh) problem and adapted to run very efficiently in IBM Cell B.E. processors [49]. The problem has also been programmed for GPGPUs accelerators using our platform *Waris* [50]. In both cases, they run orders of magnitude faster than the Alyas case. They are based on Cartesian meshes and finite differences schemes with a pre-process stage to embed heart geometry.

A second research line is to advance further the coupling of the activation potential with the mechanical deformation. In [39, 40] we present an electromechanical model, which uses a large strain total Lagrangian formulation as the mechanical model. The material model implemented is an Ogden type specially designed for the heart. Both, the electrical and the mechanical models, are integrated in Alya, keeping intact the good scalability properties of the scheme presented here, now for the coupled model.

ACKNOWLEDGEMENTS

P. Villar was awarded with one of the International L'Oreal-UNESCO 'For Women in Science' Fellowship to take part in this research project.

REFERENCES

1. Torrent-Guasp F, Kocica MJ, Corno AF, Komeda M, Carreras-Costa F, Flotats A, Cosin-Aguillar J, Wen H. Towards new understanding of the heart structure and function. *European Journal of Cardiothoracic Surgery* 2005; **27**:191–201.
2. Streeter DD. Gross morphology and fiber geometry of the heart. In *The Handbook of Physiology Section 2. The Heart* (*American Physiology Society*), Berne RM, Sperelakis N (eds), vol. 1. Williams and Wilkins: Baltimore, 1979; 61–112.
3. Bordas R, Carpentieri B, Fotia G, Maggio F, Nobes R, Pitt-Francis J, Southern J. Simulation of cardiac electrophysiology on next-generation high-performance computers. *Philosophical Transactions of the Royal Society of London* 2009; **367**:1951–1969.
4. Hunter PJ, Crampin EJ, Nielsen PMF. Bioinformatics, multiscale modeling and the IUPS Physiome Project. *Briefings in Bioinformatics* 2008; **9**(4):333–343.
5. LeGrice I, Hunter P, Young A, Smaill B. The architecture of the heart: a data based model. *Philosophical Transactions of the Royal Society of London* 2001; **359**:1217–1232.
6. Gil D, Garcia-Barnes J, Vázquez M, Arís R, Houzeaux G. Patient-sensitive anatomic and functional 3D model of the left ventricle. Presented at *the Eighth World Congress on Computational Mechanics* (*WCCM8*), *Fifth European Congress on Computational Methods in Applied Sciences and Engineeering* (*ECCOMAS 2008*), Venice, Italy, 2008.

DOI: 10.1002/cnm

7. Vázquez M, Arís R, Rosolen A, Houzeaux G. A high performance computational electrophysiology model. Presented at *the Eighth World Congress on Computational Mechanics* (*WCCM8*), *Fifth European Congress on Computational Methods in Applied Sciences and Engineeering* (*ECCOMAS 2008*), Venice, Italy, 2008.
8. Belik ME, Usyk TP, McCulloch AD. Computational methods for cardiac electrophysiology. In *Computational Models for the Human Body*, Ciarlet PG, Ayache N (eds). Handbook of Numerical Analysis, vol. XII. Elsevier: Amsterdam, 2004; 129–187.
9. Winslow RL, Scollan DF, Holmes A, Yung CK, Zhang J, Jafri MS. Electrophysiological modeling of cardiac ventricular function: from cell to organ. *Annual Review of Biomedical Engineering* 2000; **2**:119–155.
10. McCulloch AD, Huber G. Integrative biological modeling in silico. In *Silico Simulation of Biological Processes*, *Novartis Foundation Symposium 247*, Bock G, Goode JA (eds). Wiley: New York, 2002; 4–19.
11. Murillo M, Cai XC. A fully implicit parallel algorithm for simulating the non-linear electrical activity of the heart. *Numerical Linear Algebra with Applications* 2004; **11**:261–277.
12. Fenton FH, Cherry EM, Karma A, Rappel W. Modeling wave propagation in realistic heart geometries using the phase-field method. *Chaos* 2005; **15**:013502.
13. Alya System Web site. Available from: http://www.bsc.es/.
14. Houzeaux G, Vázquez M, Aubry R, Cela JM. A massively parallel fractional step solver for incompressible flows. *Journal of Computational Physics* 2009; **228**:6316–6332.
15. Houzeaux G, Vázquez M, Saez X, Cela JM. Hybrid MPI-OpenMP performance in massively parallel computational fluid dynamics. Presented at *the PARCFD 2008 International Conference on Parallel Computational Fluid Dynamics*, Lyon, Francia, 2008.
16. Houzeaux G, Aubry R, Vázquez M. Extension of fractional step techniques for incompressible flows: the preconditioned Orthomin(1) for the pressure Schur complement. *Computers and Fluids* 2011; **44**(1):297–313. DOI: 10.1016/j.compfluid.2011.01.017.
17. Rosolen A, Ordás S, Vázquez M. Numerical schemes for the simulation of three-dimensional cardiac electrical propagation inpatient-specific ventricular geometries. *Proceedings of the European Conference on Computational Fluid Dynamics ECCOMAS CFD 2006*, Egmond aan Zee, The Netherlands, 2006.
18. Graf IM. Electrical stimulation of the human left ventricle. *Ph.D. Thesis*, University of Karlsruhe, 2005.
19. Hunter PJ, Kohl P, Noble D. Integrative models of the heart: achievements and limitations. *Philosophical Transactions of the Royal Society of London* 2001; **359**.
20. Ayache N. Computational models for the human body. In *Handbook of Numerical Analysis*, Ciarlet PG, Ayache N (eds), vol. XII. Elsevier: Amsterdam, 2004.
21. Blanc O. A computer model of human atrial arrhythmia. *Ph.D. Thesis*, École Polytechnique Fédérale de Laussane, 2002.
22. Sachse FB. *Computational Cardiology*: *Modeling of Anatomy*, *Electrophysiology*, *and Mechanics*, Lecture Notes in Computer Science, vol. 2966. Springer: Berlin, 2004.
23. Penland RC, Harrild DM, Henriquez CS. Modeling impulse propagation and extracellular potential distributions in anisotropic cardiac tissue using a finite volume element discretization. *Computing and Visualization in Science* 2002; **4**:215–226.
24. Simelius K, Nenonen J, Horacek BM. Simulation of anisotropic propagation in the myocardium with a hybrid bidomain model. In *Functional Imaging and Modeling of the Heart*, Katila T, Magnin IE, Clarysse P (eds). Lecture Notes in Computer Science, vol. 2230. Springer: Berlin, 2001.
25. FitzHugh RA. Impulses and physiological states in theoretical models of nerve membrane. *Biophysical Journal* 1961; **1**:445–466.
26. Fenton FH, Karma A. Vortex dynamics in three-dimensional continuous myocardium with fiber rotation: filament instability and fibrillation. *Chaos* 1998; **8**(1):20–47.
27. Eriksson K, Estep D, Hansbo P, Johnson C. *Computational Differential Equations*. Cambridge University Press: Cambridge, 1996.
28. Aubry R, Mut F, Lohner R. Deflated preconditioned conjugate gradient solvers for the pressure-Poisson equation. *Journal of Computational Physics* 2008; **227**(24):10196–10208.
29. Codina R. A discontinuity capturing crosswind dissipation for the finite element solution of the convection diffusion equation. *Computer Methods in Applied Mechanics and Engineering* 1993; **110**:325–342.
30. Nithiarasu P, Zienkiewicz OC, Satya Sai BVK, Morgan K, Codina R, Vázquez M. Shock capturing viscosities for the general fluid mechanics algorithm. *International Journal for Numerical Methods in Fluids* 1998; **28**:1325–1353.
31. Vázquez M. Numerical modeling of compressible laminar and turbulent flow: the CBS algorithm. *Ph.D. Thesis*, Universitat Politècnica de Catalunya, Spain, 1999.
32. Tetgen Web site. Available from: http://tetgen.berlios.de/.
33. Helm PA. Ex vivo 3D diffusion tensor imaging and quantification of cardiac laminar structure. *Magnetic Resonance in Medicine* 2005; **54**(4):850–859.
34. Castella M, Buckberg GD, Saleh S, Gharib M. Structure function interface with sequential shortening of basal and apical components of the myocardial band. *European Journal of Cardiothoracic Surgery* 2005; **27**:980–987.
35. Taccardi B. Effect of myocardial fiber direction on epicardial potentials. *Circulation* 2008; **90**:3076–3090.
36. Bihan DL, Mangin JF, Poupon C, Clark CA, Pappata S, Molko N, Chabriat H. Diffusion tensor imaging: concepts and applications. *Journal of Magnetic Resonance Imaging* 2001; **13**(4):534–546.

DOI: 10.1002/cnm

37. Rubart M, Zipes DP. Genesis of cardiac arrhytmias: Electrophysiological considerations. In *Heart Disease*, Braunwald E, Zipes DP, Libby P (eds), vol. XII. W.B. Saunders Company: Philadelphia, 2001.
38. Coghlan HC, Coghlan AR, Buckberg GD, Cox JL. The electrical spiral of the heart: its role in the helical continuum. The hypothesis of the anisotropic conducting matrix. *European Journal of Cardiothoracic Surgery* 2006; **29**:178–187.
39. Vázquez M, Arís R, Lafortune P, Villar P, Houzeaux G, Gil D, Garcia-Barnes J, Ballester M, Carreras F. Massively parallel electromechanical model of the heart. Presented at *The CMBBE Conference*, Valencia, Spain, 2010.
40. Lafortune P, Arís R, Vázquez M, Houzeaux G, Jérusalem A. Large scale continuum electromechanical cardiac simulations. Presented at *The CMBE2011 Conference*, Washington, DC, U.S.A., 2011.
41. Metis Web site. Available from: http://glaros.dtc.umn.edu/gkhome/views/metis.
42. Brucker P. *Scheduling Algorithms*. Springer: Berlin, 2003.
43. Ballester-Rodes M, Flotats A, Torrent-Guasp F, Ballester-Alomar M, Carreras F, Ferreira A, Narula J. Base-to-apex ventricular activation: Fourier studies in 29 normal individuals. *European Journal of Nuclear Medicine and Molecular Imaging* 2005; **32**:1481–1481.
44. Nickerson D, Smith N, Hunter P. New developments in a strongly coupled cardiac electromechanical model. *Europace* 2005; **7**:S118–S127.
45. Pathmanathan P, Bernabeu MO, Bordas R, Cooper J, Garny A, Pitt-Francis JM, Whiteley JP, Gavaghan DJ. A numerical guide to the solution of the bidomain equations of cardiac electrophysiology. *Progress in Biophysics and Molecular Biology* 2010; **102**:136–155.
46. Bernabeu MO, Wallman M, Rodríguez B. Shock-induced arrhythmogenesis in the human heart: a computational modelling study. *The 32nd Annual International Conference of the IEEE EMBS*, Buenos Aires, Argentina, 2010.
47. Chaste Web site. Available from: http://www.comlab.ox.ac.uk/chaste.
48. Hosoi A, Washio T, Kadooka Y, Nakajima K, Hisada T. A multi-scale heart simulation on massively parallel computers. *Proceedings of the ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis*, New Orleans, U.S.A., 2010.
49. Rubio F, Hanzich M, Arís R, Houzeaux G, Vázquez M. Parallel computational electrophysiology in cell/B.E. processors. Presented at *COMPBIOMED Conference*, Swansea, U.K., 2009.
50. Rubio F, Hanzich M, Arís R, Vázquez M, Houzeaux G. Parallel Computational Electrophysiology in NVIDIA GPUs. Presented at *The Second COMPBIOMED Conference*, Washington, U.S.A., 2011.

DOI: 10.1002/cnm