A Multiresolution Space-Time Adaptive Scheme for the Bidomain Model in Electrocardiology

Mostafa Bendahmane,[1] **Raimund Bürger,**[2] **Ricardo Ruiz-Baier**[3]

[1] *Department of Mathematics, Al-Imam University, Saudi Arabia*

[2] *CI^{2}MA and Departamento de Ingeniería Matemática, Facultad de Ciencias Físicas y Matemáticas, Universidad de Concepción, Casilla 160-C, Concepción, Chile*

[3] *IACS, Chair of Modelling and Scientific Computing, École Polytechnique Fédérale de Lausanne, Station 8, CH-1015 Lausanne, Switzerland*

Received 29 September 2008; accepted 22 April 2009
Published online 29 October 2009 in Wiley Online Library (wileyonlinelibrary.com).
DOI 10.1002/num.20495

The bidomain model of electrical activity of myocardial tissue consists of a possibly degenerate parabolic PDE coupled with an elliptic PDE for the transmembrane and extracellular potentials, respectively. This system of two scalar PDEs is supplemented by a time-dependent ODE modeling the evolution of the gating variable. In the simpler subcase of the monodomain model, the elliptic PDE reduces to an algebraic equation. Since typical solutions of the bidomain and monodomain models exhibit wavefronts with steep gradients, we propose a finite volume scheme enriched by a fully adaptive multiresolution method, whose basic purpose is to concentrate computational effort on zones of strong variation of the solution. Time adaptivity is achieved by two alternative devices, namely locally varying time stepping and a Runge-Kutta-Fehlberg-type adaptive time integration. A series of numerical examples demonstrates that these methods are efficient and sufficiently accurate to simulate the electrical activity in myocardial tissue with affordable effort. In addition, the optimal choice of the threshold for discarding nonsignificant information in the multiresolution representation of the solution is addressed, and the numerical efficiency and accuracy of the method is measured in terms of CPU time speed-up, memory compression, and errors in different norms. © 2009 Wiley Periodicals, Inc. Numer Methods Partial Differential Eq 26: 1377–1404, 2010

Keywords: bidomain model; electrocardiology; fully adaptive multiresolution schemes; local time stepping; parabolic-elliptic system; Runge-Kutta-Fehlberg

Correspondence to: Raimund Bürger, CI^2 MA and Departamento de Ingeniería Matemática, Facultad de Ciencias Físicas y Matemáticas, Universidad de Concepción, Casilla 160-C, Concepción, Chile (e-mail: rburger@ing-mat.udec.cl)

Contract grant sponsor: Fondecyt; contract grant numbers: 1070682 and 1090456
Contract grant sponsor: Fondap; contract grant number: 15000001
Contract grant sponsor: BASAL project CMM, Universidad de Chile and Centro de Investigación en Ingeniería Matemática (CI^{2}MA), Universidad de Concepción
Contract grant sponsor: ERC; contract grant number: AG227058

I. INTRODUCTION

A. Scope

The obvious difficulty of direct measurements in electrocardiology early motivated interest in the numerical simulation of cardiac models. In 1952, Hodgkin and Huxley [1] introduced the first mathematical model of wave propagation in squid nerve, which was modified later on to describe several phenomena in biology, including the first physiological model of cardiac tissue (see [2]). Among these models, the *bidomain model*, first introduced by Tung [3], is one of the most accurate and complete models for the theoretical and numerical study of the electric activity in cardiac tissue. The bidomain equations result from the principle of conservation of current between the intra- and extracellular domains, followed by a homogenization process (see e.g. [2, 4, 5]) derived from a scaled version of a cellular model on a periodic structure of cardiac tissue. Mathematically, the bidomain model is a system consisting of a scalar, possibly degenerate parabolic PDE coupled with a scalar elliptic PDE for the transmembrane potential and the extracellular potential, respectively. These equations are supplemented by a time-dependent ODE for the *gating variable*, which is defined at every point of the spatial computational domain. The term "bidomain" reflects that, in general, the intra- and extracellular tissues have different longitudinal and transversal (with respect to the fiber) conductivities; if these are equal, then the model is termed *monodomain model*, and the elliptic PDE reduces to an algebraic equation. The degenerate nature of the bidomain model is essentially due to the differences between the intra- and extracellular anisotropy of the cardiac tissue [4, 6].

The bidomain model is a challenge for computation since the width of an excitation front is roughly two orders of magnitude smaller than the long axis of a human-size right ventricle. This feature of localization, along with strongly varying time scales in the reaction terms, produces solutions with sharp propagating wave fronts in the potential field, which almost precludes simulations with uniform grids. Clearly, cardiac simulations should be based on space- (and also time-) adaptive methods.

It is the purpose of this article to advance a fully adaptive multiresolution (MR) scheme with locally varying space-time stepping (LTS) and adaptive time step control by means of a Runge-Kutta-Fehlberg (RKF) method for the numerical solution of the bidomain and monodomain models. The LTS and RKF strategies are of different nature, but may be combined to obtain a potentially more powerful method (as suggested, e.g., in [7]; however, herein we do not pursue this). We furthermore deduce an optimal threshold value for discarding nonsignificant data, which permits to achieve significant data compression. This deduction motivated by the rigorous analysis performed in [8] for scalar conservation laws in one space dimension and in [9] for strictly parabolic PDEs. Experience with degenerate parabolic equations and reaction–diffusion systems [9–12] suggests that the MR method should provide an efficient tool for solving the bidomain equations. Consequently, we herein extend this method to the novel application of the bidomain and monodomain models.

The efficiency of the MR method is a consequence of the fact that, at each time step, the solution is encoded with respect to a MR basis corresponding to a hierarchy of nested grids. The size of the details determines the level of refinement needed to obtain an accurate local representation of the solution. Therefore, an adaptive mesh is evolved in time by refining and coarsening in a suitable way, by means of a strategy based on the prediction of the displacement and creation of steep gradients and similar singularities in the solution.

We apply the MR approach to an explicit finite volume (FV) method in each time step. Even though implicit methods allow larger time steps, we need to iteratively solve a nonlinear system in each time step, using e.g., Newton-Raphson method. The number of iterations is usually controlled by measuring the residual error and cannot be controlled a priori. Thus, it appears difficult to assess the true benefits of a time-stepping strategy if the basic time discretization is an implicit one, the latter being less efficient than explicit ones, especially when the overall number of time steps is large (see e.g., [13]).

B. Related Work

We first mention that standard theory for coupled parabolic–elliptic systems (see e.g., [14]) does not apply naturally to the bidomain equations, since the anisotropies of the intra- and extracellular media differ and the resulting system is of degenerate parabolic type. Colli Franzone and Savaré [6] present a weak formulation for the bidomain model and show that it has a structure suitable for applying the theory of evolution variational inequalities in Hilbert spaces. Bendahmane and Karlsen [4] prove existence and uniqueness for the bidomain equations using the Faedo-Galerkin method and compactness theory for the existence part, and Bourgault et al. [15] prove existence and uniqueness for the bidomain equations by first reformulating the problem as a single parabolic PDE and then applying a semigroup approach.

From a computational point of view, substantial contributions have been made in adaptivity for cardiac models. However, our treatment differs to the best of our knowledge from other adaptive approaches in the literature, which include adaptive mesh refinement (AMR) (e.g., [13]), adaptive finite element methods using a posteriori error techniques (see e.g., [5]) or multigrid methods applied to finite elements. Furthermore, Quan et al. [16] present a domain decomposition approach using an alternating direction implicit (ADI) method. Sundnes et al. [17] introduce an operator splitting method to solve a fully coupled discretization of three PDEs modelling the interaction between the myocardium and the torso surrounding the heart. With respect to time adaptivity, Skouibine et al. [18] present a predictor–corrector time stepping strategy to accelerate a given finite difference scheme for the bidomain equations using active membrane kinetics (Luo-Rudy phase II). Cherry et al. [13] use local time stepping, similar to the method introduced by Berger and Oliger [19], to accelerate a reference scheme. Parallelized versions of part of the methods mentioned earlier are presented, for example, by Colli Franzone and Pavarino [20] and Saleheen and Ng [21].

MR schemes for hyperbolic partial differential equations were first proposed by Harten [22]. We refer to Müller [23] for a survey on MR methods, see also Chiavassa et al. [24], and Cohen et al. [8] and Dahmen et al. [25] for the application of classical MR methods to hyperbolic partial differential equations.

The aim of the MR method is to accelerate a given reference discretization scheme while controlling the error. In the context of fully adaptive MR methods [8], the mathematical analysis is complete only in the case of a scalar conservation law, but in practice, these techniques have been used by several groups (see e.g., [7,9,10,23,26]) to successfully solve a wide class of problems, including applications to multidimensional systems. These results illustrate that the MR method has turned out to be a useful device for a series of problems with a similar structure to cardiac problems. Thus, our motivation is to point out that this versatile method also provides an efficient tool for the simulation of electrical activity in cardiac tissue.

C. Outline of the Article

The remainder of this article is organized as follows. In Section II, the bidomain and monodomain models of cardiac tissue are outlined. Section III deals with the construction of an appropriate FV method for the solution of both the parabolic–elliptic system and the reaction–diffusion equation arising from the bidomain and monodomain models, respectively. Next, in Section IV, we develop the MR analysis used to endow the reference FV schemes with space adaptivity. More precisely, in Section IVA, we introduce the wavelet basis underlying the MR representation with the pertinent projection operator. In Section IVB, the prediction operator, the detail coefficients and the thresholding procedure are introduced. In Section IVC, we recall the graded tree data structure used for storage of the numerical solution, and which is introduced for ease of navigation. In Section IVD, we first recall the results of the rigorous error analyses of Cohen et al. [8] and Roussel et al. [9] referring to conservation laws and strictly parabolic equations, respectively, and then show how this analysis motivates the choice of a reference tolerance ε_{R} for the bidomain and monodomain models in a fashion similar to the treatment of scalar degenerate parabolic equations [11, 12]. The quantity ε_{R} determines the comparison values ε_l used for the thresholding operation at each level l of resolution. Overall, the basic goal is to choose the threshold values in such a way that the resulting MR scheme has the same order of accuracy as the usual FV scheme.

In Section V, we address two strategies for the adaptive evolution in time of the space-adaptive MR scheme, namely the locally varying time stepping (LTS, Section VA) and a variant of the well-known Runge-Kutta-Fehlberg (RKF, Section VB) method. Finally, in Section VI, we present numerical examples putting into evidence the efficiency of the underlying methods. Some conclusions of the article are given in Section VII.

II. THE MACROSCOPIC BIDOMAIN AND MONODOMAIN MODELS

We consider a bounded open subset $\Omega \subset \mathbb{R}^2$ with a piecewise smooth boundary $\partial\Omega$. The domain Ω represents a slice of the cardiac muscle regarded as two interpenetrating and superimposed (anisotropic) continuous media, namely the intracellular (i) and extracellular (e) tissues. These tissues occupy the same two-dimensional area, and are separated from each other (and connected at each point) by the cardiac cellular membrane. The quantities of interest are intracellular and extracellular electric potentials, $u_{\mathrm{i}} = u_{\mathrm{i}}(x,t)$ and $u_{\mathrm{e}} = u_{\mathrm{e}}(x,t)$, at $(x,t) \in \Omega_T := \Omega \times (0,T)$. Their difference $v = v(x,t) := u_{\mathrm{i}} - u_{\mathrm{e}}$ is the transmembrane potential. The conductivity of the tissue is represented by scaled tensors $\mathbf{M}_{\mathrm{i}}(x)$ and $\mathbf{M}_{\mathrm{e}}(x)$ given by

$$\mathbf{M}_k(x) = \sigma_k^{\mathrm{t}}\mathbf{I} + \left(\sigma_k^{\mathrm{l}} - \sigma_k^{\mathrm{t}}\right)\mathbf{a}_l(x)\mathbf{a}_l^{\mathrm{T}}(x), \qquad k \in \{\mathrm{e},\mathrm{i}\},$$

where $\sigma_k^{\mathrm{l}} = \sigma_k^{\mathrm{l}}(x) \in C^1(\mathbb{R}^2)$ and $\sigma_k^{\mathrm{t}} = \sigma_k^{\mathrm{t}}(x) \in C^1(\mathbb{R}^2)$, $k \in \{\mathrm{e},\mathrm{i}\}$, are the intra- and extracellular conductivities along and transversal to the direction of the fiber (parallel to the unitary direction vector denoted by $\mathbf{a}_l(x)$), respectively, and $\mathbf{a}_l^{\mathrm{T}}(x)$ is the transpose of $\mathbf{a}_l(x)$. For fibers aligned with the axis, $\mathbf{M}_{\mathrm{i}}(x)$ and $\mathbf{M}_{\mathrm{e}}(x)$ are diagonal matrices: $\mathbf{M}_{\mathrm{i}}(x) = \mathrm{diag}(\sigma_{\mathrm{i}}^{\mathrm{l}}, \sigma_{\mathrm{i}}^{\mathrm{t}})$ and $\mathbf{M}_{\mathrm{e}}(x) = \mathrm{diag}(\sigma_{\mathrm{e}}^{\mathrm{l}}, \sigma_{\mathrm{e}}^{\mathrm{t}})$. When the anisotropy ratios $\sigma_{\mathrm{i}}^{\mathrm{l}}/\sigma_{\mathrm{i}}^{\mathrm{t}}$ and $\sigma_{\mathrm{e}}^{\mathrm{l}}/\sigma_{\mathrm{e}}^{\mathrm{t}}$ are equal, we are in the case of equal anisotropy, and the case where the conductivities in the longitudinal direction l are higher than those across the fiber (transversal direction t) is called strong anisotropy of electrical conductivity. When the fibers rotate from bottom to top, this type of anisotropy is often referred to as rotational anisotropy.

The bidomain model is given by the following coupled reaction–diffusion system (see e.g., [27]):

$$\beta c_{\mathrm{m}}\partial_t v - \nabla \cdot \big(\mathbf{M}_{\mathrm{i}}(x)\nabla u_{\mathrm{i}}\big) + \beta I_{\mathrm{ion}}(v,w) = 0, \tag{2.1a}$$

$$\beta c_{\mathrm{m}}\partial_t v + \nabla \cdot \big(\mathbf{M}_{\mathrm{e}}(x)\nabla u_{\mathrm{e}}\big) + \beta I_{\mathrm{ion}}(v,w) = I_{\mathrm{app}}, \tag{2.1b}$$

$$\partial_t w - H(v,w) = 0, \quad (x,t) \in \Omega_T. \tag{2.1c}$$

Here, $c_{\mathrm{m}} > 0$ is the surface capacitance of the membrane, β is the surface-to-volume ratio, and $w(x,t)$ is the gating or recovery variable. The stimulation currents applied to the extracellular space are represented by the function $I_{\mathrm{app}} = I_{\mathrm{app}}(x,t)$. The functions $H(v,w)$ and $I_{\mathrm{ion}}(v,w)$ correspond to the fairly simple Mitchell-Schaeffer membrane model [28] for the membrane and ionic currents:

$$H(v,w) = \frac{w_\infty(v/v_p) - w}{R_{\mathrm{m}} c_{\mathrm{m}} \eta_\infty(v/v_p)}, \qquad I_{\mathrm{ion}}(v,w) = \frac{v_p}{R_{\mathrm{m}}}\left(\frac{v}{v_p \eta_2} - \frac{v^2(1 - v/v_p)w}{v_p^2 \eta_1}\right), \tag{2.2}$$

where $\eta_\infty(s) = \eta_3 + (\eta_4 - \eta_3)\mathcal{H}(s - \eta_5)$ and $w_\infty(s) = \mathcal{H}(s - \eta_5)$, where $\mathcal{H}$ denotes the Heaviside function, R_{m} is the surface resistivity of the membrane, and v_p and $\eta_1, \dots, \eta_5$ are given parameters. A simpler choice for the membrane kinetics is the well-known FitzHugh-Nagumo model (see [2]), which is specified by

$$H(v,w) = av - bw, \quad I_{\mathrm{ion}}(v,w) = -\lambda\big(w - v(1-v)(v-\theta)\big), \tag{2.3}$$

where a, b, λ, and θ are given parameters.

We rewrite (2.1) as the strongly coupled parabolic–elliptic PDE-ODE system (see e.g., [27])

$$\beta c_{\mathrm{m}} \partial_t v + \nabla \cdot \big(\mathbf{M}_{\mathrm{e}}(x)\nabla u_{\mathrm{e}}\big) + \beta I_{\mathrm{ion}}(v,w) = I_{\mathrm{app}}, \tag{2.4a}$$

$$\nabla \cdot \big((\mathbf{M}_{\mathrm{i}}(x) + \mathbf{M}_{\mathrm{e}}(x))\nabla u_{\mathrm{e}}\big) + \nabla \cdot \big(\mathbf{M}_{\mathrm{i}}(x)\nabla v\big) = I_{\mathrm{app}}, \tag{2.4b}$$

$$\partial_t w - H(v,w) = 0, \quad (x,t) \in \Omega_T. \tag{2.4c}$$

We utilize zero flux boundary conditions corresponding to an isolated piece of cardiac tissue,

$$\big(\mathbf{M}_k(x)\nabla u_k\big) \cdot \mathbf{n} = 0 \text{ on } \Sigma_T := \partial\Omega \times (0,T), \quad k \in \{\mathrm{e}, \mathrm{i}\}, \tag{2.5}$$

and impose initial conditions (which are degenerate for the transmembrane potential v):

$$v(0,x) = v_0(x), \quad w(0,x) = w_0(x), \quad x \in \Omega. \tag{2.6}$$

We require the initial datum v_0 to be compatible with (2.5) in the following sense. If we fix both $u_k(0,x)$, $k \in \{\mathrm{e}, \mathrm{i}\}$ as initial data, the problem may become unsolvable, since the time derivative involves only $v = u_{\mathrm{i}} - u_{\mathrm{e}}$ (this is also referred as degeneracy in time). Thus, we impose the compatibility condition

$$\int_\Omega u_{\mathrm{e}}(x,t)\,dx = 0 \quad \text{for a.e. } t \in (0,T). \tag{2.7}$$

In the case that $\mathbf{M}_{\mathrm{i}} \equiv \mu \mathbf{M}_{\mathrm{e}}$ for some constant $\mu \in \mathbb{R}$, the system (2.1) is equivalent to a scalar parabolic equation for v, coupled to an ODE for w. This parabolic equation is obtained by multiplying (2.1a) by $1/(1+\mu)$, (2.1b) by $\mu/(1+\mu)$, and adding the results. The final monodomain model can be stated as follows:

$$\beta c_{\mathrm{m}} \partial_t v - \nabla \cdot \left(\frac{\mathbf{M}_{\mathrm{i}}}{1+\mu}\nabla v\right) + \beta I_{\mathrm{ion}}(v,w) = \frac{\mu}{1+\mu} I_{\mathrm{app}},$$

$$\partial_t w - H(v,w) = 0, \quad (x,t) \in \Omega_T. \tag{2.8}$$

This simpler model requires less computational effort than (2.4), and even though the assumption of equal anisotropy ratios is very strong and generally unrealistic, (2.8) is adequate for a qualitative investigation of repolarization sequences and the distribution of patterns of durations of the action potential [29].

We assume that the functions $\mathbf{M}_{\mathrm{e}}$, $\mathbf{M}_{\mathrm{i}}$, I_{ion}, and H are sufficiently smooth so that the following definitions of weak solutions make sense. Furthermore, we assume that $I_{\mathrm{app}} \in L^2(\Omega_T)$ and $\mathbf{M}_k \in L^\infty(\Omega)$ and $\mathbf{M}_k\boldsymbol{\xi} \cdot \boldsymbol{\xi} \geq C_M|\boldsymbol{\xi}|^2$ for a.e. $x \in \Omega$, for all $\boldsymbol{\xi} \in \mathbb{R}^2$, $k \in \{\mathrm{e}, \mathrm{i}\}$, and a constant $C_M > 0$. For the sake of completeness, we state the definitions of a weak solution for the bidomain and the monodomain model, respectively.

Definition 2.1. *A triple* $\mathbf{u} = (v, u_{\mathrm{e}}, w)$ *of functions is a* weak solution of the bidomain model (2.4)–(2.6) *if* $v, u_{\mathrm{e}} \in L^2(0, T; H^1(\Omega))$, $w \in C([0, T], L^2(\Omega))$, (2.7) *is satisfied, and the following identities hold for all test functions* $\varphi, \psi, \xi \in \mathcal{D}([0, T) \times \bar{\Omega})$*:*

$$\beta c_{\mathrm{m}} \int_\Omega v_0(x)\varphi(0, x)\,dx + \iint_{\Omega_T} \left\{\beta c_{\mathrm{m}} v \partial_t \varphi - \mathbf{M}_{\mathrm{e}}(x)\nabla u_{\mathrm{e}} \cdot \nabla\varphi + \beta I_{\mathrm{ion}}\varphi\right\} dx\,dt$$
$$= \iint_{\Omega_T} I_{\mathrm{app}}\varphi\,dx\,dt,$$
$$\iint_{\Omega_T} \left\{-\big(\mathbf{M}_{\mathrm{i}}(x) + \mathbf{M}_{\mathrm{e}}(x)\big)\nabla u_{\mathrm{e}} \cdot \nabla\psi - \mathbf{M}_{\mathrm{i}}(x)\nabla v \cdot \nabla\psi\right\} dx\,dt = \iint_{\Omega_T} I_{\mathrm{app}}\varphi\,dx\,dt,$$
$$-\int_\Omega w_0(x)\xi(0, x)\,dx - \iint_{\Omega_T} w\partial_t\xi\,dx\,dt = \iint_{\Omega_T} H\xi\,dx\,dt.$$

Definition 2.2. *A pair* $\mathbf{u} = (v, w)$ *of functions is a* weak solution of the monodomain model (2.8) *if* $v \in L^2(0, T; H^1(\Omega))$, $w \in C([0, T], L^2(\Omega))$, *and the following identities hold for all test functions* $\varphi, \xi \in \mathcal{D}([0, T) \times \bar{\Omega})$*:*

$$\beta c_{\mathrm{m}} \int_\Omega v_0(x)\varphi(0, x)\,dx + \iint_{\Omega_T} \left\{\beta c_{\mathrm{m}} v \partial_t \varphi + \beta I_{\mathrm{ion}}\varphi - \frac{1}{1+\mu}\mathbf{M}_{\mathrm{i}}\nabla v \cdot \nabla\varphi\right\} dx\,dt$$
$$= \frac{\mu}{1+\mu}\iint_{\Omega_T} I_{\mathrm{app}}\varphi\,dx\,dt,$$
$$-\int_\Omega w_0(x)\xi(0, x)\,dx - \iint_{\Omega_T} w\partial_t\xi\,dx\,dt = \iint_{\Omega_T} H\xi\,dx\,dt.$$

III. THE REFERENCE FINITE VOLUME SCHEME

To approximate solutions to the bidomain equations (2.4), we employ a standard FV scheme, which is described here for a uniform grid. The square spatial domain $\Omega \subset \mathbb{R}^2$ is partitioned into control volumes $(\Omega_{ij})_{1 \leq i,j \leq N}$, where we define

$$\Omega_{ij} := [x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$$

and $h := x_{i+1/2} - x_{i-1/2} = y_{j+1/2} - y_{j-1/2}$ for all $1 \leq i, j \leq N$. We also choose a time step size $\Delta t > 0$ and set $t^n := n\Delta t$ for $n \in \{0, \dots, N'\}$, where $N' > 0$ is the smallest integer such that $N'\Delta t \geq T$. The cell average of a quantity q over Ω_{ij} at time t^n is defined by

$$q_{ij}^n := \frac{1}{h^2} \iint_{\Omega_{ij}} q(x, t^n)\, dx\, dt.$$

We define the unknowns

$$H_{ij}^n := H\big(v_{ij}^n, w_{ij}^n\big), \quad I_{\mathrm{ion},(i,j)}^n := I_{\mathrm{ion}}\big(v_{ij}^n, w_{ij}^n\big), \tag{3.1}$$

and the cell averages of the given function I_{app} at time t^n:

$$I_{\mathrm{app},(i,j)}^n := \frac{1}{h^2} \iint_{\Omega_{ij}} I_{\mathrm{app}}(x, t^n)\, dx\, dt.$$

Denoting by θ the angle of alignment of the fibers, we may recast $\mathbf{M}_k(x)$ explicitly in the form (see e.g. [30])

$$\mathbf{M}_k = \begin{bmatrix} \sigma_k^{\mathrm{t}} + (\sigma_k^{\mathrm{l}} - \sigma_k^{\mathrm{t}}) \sin^2(\theta) & (\sigma_k^{\mathrm{l}} - \sigma_k^{\mathrm{t}}) \sin(\theta)\cos(\theta) \\ (\sigma_k^{\mathrm{l}} - \sigma_k^{\mathrm{t}}) \sin(\theta)\cos(\theta) & \sigma_k^{\mathrm{t}} + (\sigma_k^{\mathrm{l}} - \sigma_k^{\mathrm{t}}) \cos^2(\theta) \end{bmatrix} \quad \text{for } k \in \{\mathrm{e}, \mathrm{i}\}.$$

Therefore, defining the difference operators $\delta_x^s V_{ij} := V_{i+s,j} - V_{ij}$ and $\delta_y^s V_{ij} := V_{i,j+s} - V_{ij}$, we define the numerical fluxes (we only provide those corresponding to the flux $(i+1, j) \to (i, j)$)

$$F_{(i+1,j)\to(i,j)}^{u,\mathrm{e}} := \frac{1}{h} M_{\mathrm{e}}^{11} \delta_x^1 u_{\mathrm{e},(i,j)},$$

$$F_{(i+1,j)\to(i,j)}^{u,\mathrm{i,e}} := \frac{1}{h} \left(M_{\mathrm{e}}^{11} + M_{\mathrm{i}}^{11}\right) \delta_x^1 u_{\mathrm{e},(i,j)},$$

$$F_{(i+1,j)\to(i,j)}^{v,\mathrm{i}} := \frac{1}{h} M_{\mathrm{i}}^{11} \delta_x^1 v_{ij},$$

and following e.g., [21], the corresponding anisotropic diffusive term may be approximated by

$$\nabla \cdot (\mathbf{M}_k \nabla u)_{i,j} \approx \mathcal{D}_h(\mathbf{M}_k, u_{ij}) := \frac{1}{h^2} \sum_{s \in \{-1,1\}} M_k^{11} \delta_x^s u_{ij} + M_k^{12} u_{i+s,j+s} + M_k^{21} u_{i+s,j-s} + M_k^{22} \delta_y^s u_{ij}$$

for all $1 \le i, j \le N$ and $k \in \{\mathrm{e}, \mathrm{i}\}$. We now describe the finite volume scheme employed to advance the numerical solution from t^n to t^{n+1}, which is based on a simple explicit Euler time discretization. The computation starts from the initial cell averages

$$v_{ij}^0 = \frac{1}{h^2} \int_{\Omega_{ij}} v_0(x)\, dx, \quad w_{ij}^0 = \frac{1}{h^2} \int_{\Omega_{ij}} w_0(x)\, dx. \tag{3.2}$$

Now, we first integrate the corresponding equations, average over Ω_{ij} and discretize. Then, assuming that at $t = t^n$ the quantities $u_{k,(i,j)}^n$, $k \in \{\mathrm{e}, \mathrm{i}\}$, v_{ij}^n and w_{ij}^n are known for all Ω_{ij}, we compute their new values at $t = t^{n+1}$, $u_{k,(i,j)}^{n+1}$, $k \in \{\mathrm{e}, \mathrm{i}\}$, v_{ij}^{n+1} and w_{ij}^{n+1}, from

$$\beta c_{\mathrm{m}} \frac{v_{ij}^{n+1} - v_{ij}^n}{\Delta t} + \mathcal{D}_h\big(\mathbf{M}_{\mathrm{e}}, u_{\mathrm{e},(i,j)}^n\big) + \beta I_{\mathrm{ion},(i,j)}^n = I_{\mathrm{app},(i,j)}^n, \tag{3.3}$$

$$\mathcal{D}_h\big((\mathbf{M}_{\mathrm{e}} + \mathbf{M}_{\mathrm{i}}), u_{\mathrm{e},(i,j)}^{n+1}\big) + \mathcal{D}_h\big(\mathbf{M}_{\mathrm{i}}, v_{ij}^{n+1}\big) = I_{\mathrm{app},(i,j)}^n, \tag{3.4}$$

$$\frac{w_{ij}^{n+1} - w_{ij}^n}{\Delta t} - H_{ij}^n = 0. \tag{3.5}$$

The boundary condition (2.5) is taken into account by imposing zero fluxes on the external edges (which we only show here for the right boundary of a square domain):

$$F^{u,\mathrm{e}}_{(i,N)\to(i,N-1)} \equiv \frac{1}{h} M_{\mathrm{e}}^{11} \delta_x^1 u_{\mathrm{e},(i,N-1)} = 0,$$

$$F^{u,\mathrm{i,e}}_{(i,N)\to(i,N-1)} \equiv \frac{1}{h} \left(M_{\mathrm{e}}^{11} + M_{\mathrm{i}}^{11}\right) \delta_x^1 u_{\mathrm{e},(i,N-1)} = 0$$

for $1 \le i \le N-1$, and the compatibility condition (2.7) is discretized via

$$\sum_{i,j=1}^{N} h^2 u^n_{\mathrm{e},(i,j)} = 0, \quad n = 0,1,2,\dots.$$

Analogously, a FV method for the monodomain model (2.8) is given by (3.2) and the following formulas to advance the solution over one time step:

$$\beta c_{\mathrm{m}} \frac{v_{ij}^{n+1} - v_{ij}^n}{\Delta t} + \mathcal{D}_h\left(\frac{1}{1+\mu}\mathbf{M}_{\mathrm{i}}, v_{ij}^n\right) + \beta I^n_{\mathrm{ion},(i,j)} = \frac{\mu}{1+\mu} I^n_{\mathrm{app},(i,j)},$$

$$\frac{w_{ij}^{n+1} - w_{ij}^n}{\Delta t} - H_{ij}^n = 0.$$

A FV method on arbitrary meshes for a different version of the bidomain equations is analyzed in [31]. In that article, the authors prove existence and uniqueness of solutions to an implicit FV scheme, and provide convergence results. On the other hand, following [31,32], we prove in [33] existence and uniqueness of approximate solutions (that is, well-definedness of the scheme) for an implicit FV method on rectangular meshes, and show that it converges to the corresponding weak solution for the monodomain problem, and also for the bidomain equations in the special case when $\mathbf{M}_{\mathrm{i}}$ and $\mathbf{M}_{\mathrm{e}}$ are diagonal tensors.

Moreover, as in [10], we may deduce that the explicit version of the FV method used herein, (3.2)–(3.5), is stable under the CFL condition

$$\Delta t \le h\left(2 \max_{1\le i,j\le N}\left(\left|I^n_{\mathrm{ion},(i,j)}\right| + \left|I^n_{\mathrm{app},(i,j)}\right|\right) + 4h^{-1} \max_{1\le i,j\le N}\left(\left|M_{\mathrm{i},(i,j)}\right| + \left|M_{\mathrm{e},(i,j)}\right|\right)\right)^{-1}. \tag{3.6}$$

Note that the values of $I^n_{\mathrm{ion},(i,j)}$ and $I^n_{\mathrm{app},(i,j)}$ depend on time. Since I_{app} is a given control function, $\max_{1\le i,j\le N} |I^n_{\mathrm{app},(i,j)}|$ can assumed to be bounded, but $I^n_{\mathrm{ion},(i,j)}$ is not bounded a priori for arbitrarily large times. Consequently, in our computations, we evaluate the right-hand side of (3.6) after each iteration at $t = t^n$, and use (3.6) to define the time step size Δt to advance the solution from t^n to $t^{n+1} = t^n + \Delta t$.

The following algorithm shows how the solution $\mathbf{u}^{n+1} = (v, u_{\mathrm{e}}, w)^{n+1}$ is obtained in each time step

Algorithm 3.1 Time discretization.

1. *Assume that u_{i}^n, u_{e}^n, v^n and w^n are known (at time t^n).*
2. *Compute w^{n+1} using*

$$\frac{w^{n+1} - w^n}{\Delta t} - H(v^n, w^n) = 0.$$

3. *Calculate v^{n+1} by solving*

$$\beta c_{\mathrm{m}} \frac{v^{n+1} - v^n}{\Delta t} + \mathcal{D}_h\big(\mathbf{M}_{\mathrm{e}}, u_{\mathrm{e}}^n\big) + \beta I_{\mathrm{ion}}(v^n, w^n) = I_{\mathrm{app}}^n.$$

4. *Calculate u_{e}^{n+1} from the linear system*

$$\mathcal{D}_h\big((\mathbf{M}_{\mathrm{e}} + \mathbf{M}_{\mathrm{i}}), u_{\mathrm{e}}^{n+1}\big) + \mathcal{D}_h\big(\mathbf{M}_{\mathrm{i}}, v^{n+1}\big) = I_{\mathrm{app}}^n.$$

This algorithm structure is usually preferred for systems involving parabolic and elliptic equations, since it explicitly isolates the solution of the elliptic problem from the rest of the computations [34].

IV. MULTIRESOLUTION AND WAVELETS

A. Wavelet Basis

Consider a rectangle which after a change of variables can be regarded as $\Omega = [0,1]^2$. We determine a nested mesh hierarchy $\Lambda_0 \subset \cdots \subset \Lambda_L$, using a uniform dyadic partition of Ω. Here each grid $\Lambda_l := \{V_{(i,j),l}\}_{(i,j)}$, with (i,j) to be defined, is formed by the control volumes on each level

$$\begin{aligned} V_{(i,j),l} &:= \big[2^{-l}i, 2^{-l}(i+1)\big] \times \big[2^{-l}j, 2^{-l}(j+1)\big], \\ i, j &\in I_l = \{0, \ldots, 2^l - 1\}, \quad l = 0, \ldots, L. \end{aligned}$$

where $l = 0$ corresponds to the coarsest and $l = L$ to the finest level, which is fixed and chosen large enough at the beginning of the process. The nestedness of the grid hierarchy is made precise by the refinement sets $\mathcal{M}_{(i,j),l} = \{2(i,j) + \mathbf{e}\}$, $\mathbf{e} \in E := \{0,1\}^2$, which satisfy $\#\mathcal{M}_{(i,j),l} = 4$. For $x = (x_1, x_2) \in V_{(i,j),l}$ the scale box function is defined as

$$\tilde{\varphi}_{(i,j),l}(x) := \frac{1}{|V_{(i,j),l}|} \chi_{V_{(i,j),l}}(x) = 2^{2l} \chi_{[0,1]^2}(2^l x_1 - i, 2^l x_2 - j),$$

and the average of any function $u(\cdot, t) \in L^1(\Omega)$ for the cell $V_{(i,j),l}$ may be expressed as the inner product

$$u_{(i,j),l} := \big\langle u, \tilde{\varphi}_{(i,j),l} \big\rangle_{L^1(\Omega)}.$$

We are now ready to define the following two-level relation for cell averages and box functions:

$$\begin{aligned} \tilde{\varphi}_{(i,j),l} &= \sum_{\mathbf{r} \in \mathcal{M}_{(i,j),l}} \frac{|V_{\mathbf{r},l+1}|}{|V_{(i,j),l}|} \tilde{\varphi}_{\mathbf{r},l+1} = \frac{1}{4} \sum_{(p,q) \in E} \tilde{\varphi}_{(2i+p,2j+q),l+1}, \\ u_{(i,j),l} &= \sum_{\mathbf{r} \in \mathcal{M}_{(i,j),l}} \frac{|V_{\mathbf{r},l+1}|}{|V_{(i,j),l}|} u_{\mathbf{r},l+1} = \frac{1}{4} \sum_{(p,q) \in E} u_{(2i+p,2j+q),l+1}, \end{aligned} \tag{4.1}$$

which defines a projection operator, which allows us to move from finer to coarser levels. For $x \in V_{2(i,j)+\mathbf{a},l+1}$ with $\mathbf{a} \in E$, we define the wavelet function depending on the box functions on a finer level

$$\tilde{\psi}_{(i,j),\mathbf{e},l} := \sum_{\mathbf{a} \in E} 2^{-2} (-1)^{\mathbf{a}\cdot\mathbf{e}} \tilde{\varphi}_{2(i,j)+\mathbf{a},l+1} = \sum_{\mathbf{r} \in \mathcal{M}_{(i,j),l}} \frac{|V_{\mathbf{r}=2(i,j)+\mathbf{a},l+1}|}{|V_{(i,j),l}|} (-1)^{\mathbf{a}\cdot\mathbf{e}} \tilde{\varphi}_{\mathbf{r},l+1}.$$

The number of related wavelets is $\#\mathcal{M}_{(i,j),l} - 1 = 3$. Since $\mathbf{r} \cdot \mathbf{e} \in \{0, 1, 2\}$ for $\mathbf{r}, \mathbf{e} \in E$, we have for instance that

$$\tilde{\psi}_{(i,j),(1,0),l} = \frac{1}{4}\left(\tilde{\varphi}_{2(i,j),l+1} + \tilde{\varphi}_{2(i,j)+(0,1),l+1} - \tilde{\varphi}_{2(i,j)+(1,0),l+1} - \tilde{\varphi}_{2(i,j)+(1,1),l+1}\right).$$

Doing this for all $\mathbf{e} \in E^* := E \setminus \{(0,0)\}$ yields an inverse two-level relation (see [23]), namely

$$\tilde{\varphi}_{2(i,j)+\mathbf{a},l+1} = \sum_{\mathbf{e}\in E} (-1)^{\mathbf{a}\cdot\mathbf{e}} \tilde{\psi}_{(i,j),\mathbf{e},l}, \quad \mathbf{a} \in E.$$

This equation is related to the concept of stable completions [23]. Roughly speaking, the L^∞-counterparts of the wavelet functions $\{\tilde{\psi}_{(i,j),l}\}_{i,j\in I_l}$ form a completion of the L^∞-counterpart of the basis system $\{\tilde{\varphi}_{(i,j),l}\}_{i,j\in I_l}$, and this determines the existence of a biorthogonal system.

B. Detail Coefficients

For $\mathbf{e} \in E^*$, we introduce the details, which will be crucial to detect zones with steep gradients:

$$d_{(i,j),\mathbf{e},l} := \langle u, \tilde{\psi}_{(i,j),\mathbf{e},l} \rangle.$$

These detail coefficients also satisfy a two-level relation, namely

$$d_{(i,j),\mathbf{e},l} = \frac{1}{4} \sum_{2(i,j)+\mathbf{a}\in\mathcal{M}_{(i,j),l}} (-1)^{\mathbf{a}\cdot\mathbf{e}} u_{2(i,j)+\mathbf{a},l+1}. \tag{4.2}$$

An appealing feature is that we can determine a transformation between the cell averages on level L and the cell averages on level zero plus a series of details. This can be achieved by applying recursively the two-level relations (4.1) and (4.2); but, we also require this transformation to be reversible:

$$\tilde{u}_{(i,j),l+1} = \sum_{\mathbf{r}\in\bar{S}^l_{(i,j)}} g^l_{(i,j),\mathbf{r}} u_{\mathbf{r},l}, \quad \bar{S}^l_{(i,j)} := \left\{V_{([i/2]+r_1,[j/2]+r_2),l}\right\}_{r_1,r_2\in\{-s,\dots,0,\dots,s\}}, \tag{4.3}$$

where $\bar{S}^l_{(i,j)}$ is the stencil of interpolation or coarsening set, $g^l_{(i,j),\mathbf{r}}$ are coefficients, and the tilde over u in the left-hand side of (4.3) means that this quantity corresponds to a predicted value.

Relation (4.3) defines the so-called prediction operator, which allows us to move from coarser to finer resolution levels. In contrast to the projection, the prediction operator is not unique, but we will impose two constraints: to be consistent with the projection, in the sense that the prediction operator is the right inverse of the projection operator, and to be local, in the sense that the predicted value depends only on $\bar{S}^l_{(i,j)}$. For the sake of notation, in our case we may write (4.3) as

$$\tilde{u}_{(2i+e_1,2j+e_2),l+1} = u_{(i,j),l} - (-1)^{e_1} Q_x - (-1)^{e_2} Q_y + (-1)^{e_1 e_2} Q_{xy},$$

where $e_1, e_2 \in \{0, 1\}$ and

$$Q_x := \sum_{n=1}^{s} \tilde{\gamma}_n \big(u_{(i+n,j),l} - u_{(i-n,j),l}\big), \quad Q_y := \sum_{p=1}^{s} \tilde{\gamma}_p \big(u_{(i,j+p),l} - u_{(i,j-p),l}\big),$$

$$Q_{xy} := \sum_{n=1}^{s} \tilde{\gamma}_n \sum_{p=1}^{s} \tilde{\gamma}_p \big(u_{(i+n,j+p),l} - u_{(i+n,j-p),l} - u_{(i-n,j+p),l} + u_{(i-n,j-p),l}\big).$$

Here the corresponding coefficients are $\tilde{\gamma}_1 = -\frac{22}{128}$ and $\tilde{\gamma}_2 = \frac{3}{128}$ (see [9]).

From [25], we know that details are related to the regularity of a given function: if u is sufficiently smooth, then its detail coefficients decrease when going from coarser to finer levels:

$$\left|d^u_{(i,j),l}\right| \leq C2^{-2lr}\left\|\nabla^{(r)}u\right\|_{L^\infty(V_{(i,j),l})},$$

where $r = 2s + 1$ is the number of vanishing moments of the wavelets. This means that the more regular u is over $V_{(i,j),l}$, the smaller is the corresponding detail coefficient, so it is natural to attempt to compress data by discarding the information corresponding to small details. This thresholding procedure basically consists in discarding all control volumes corresponding to details that are smaller in absolute value than a level-dependent tolerance ε_l,

$$\left|d^u_{(i,j),l}\right| < \varepsilon_l, \quad l = 0,\dots,L.$$

Given a reference tolerance ε_{R}, whose choice is motivated in Section D, we determine ε_l by

$$\varepsilon_l = 2^{2(l-L)}\varepsilon_{\mathrm{R}}, \quad l = 0,\dots,L. \tag{4.4}$$

For multicomponent solutions, there are many possible definitions for a scalar detail $d_{(i,j),l}$ that is calculated from the details of the components (see a brief discussion in [11]). To guarantee that the refinement and coarsening procedures are always on the safe side, in the sense that we always prefer to keep a node with a detail triple containing at least one component above the threshold (4.4), we will use

$$d^{\mathbf{u}}_{(i,j),l} = \min\left\{\left|d^v_{(i,j),l}\right|, \left|d^{u_{\mathrm{e}}}_{(i,j),l}\right|, \left|d^w_{(i,j),l}\right|\right\}$$

and

$$d^{\mathbf{u}}_{(i,j),l} = \max\left\{\left|d^v_{(i,j),l}\right|, \left|d^{u_{\mathrm{e}}}_{(i,j),l}\right|, \left|d^w_{(i,j),l}\right|\right\}$$

for the refinement and coarsening procedures, respectively. In practice, these details are computed simply as the differences between the "exact" and the predicted value:

$$d^{\mathbf{u}}_{(i,j),l} := \mathbf{u}_{(i,j),l} - \hat{\mathbf{u}}_{(i,j),l}.$$

C. Graded Tree Data Structure

We organize the cell averages and corresponding details at different levels in a dynamic graded tree (see Fig. 1): whenever a node is included in the tree, all other nodes corresponding to the same spatial region in coarser resolution levels are also included, and neighboring cells will differ by at most one refinement level. This choice guarantees the stability of the multilevel operations [8]. We denote by *root* the basis of the tree. In two space dimensions, a parent node has four sons, and the sons of the same parent are called brothers. A node without sons is called a leaf. A given node has $s' = 2$ nearest neighbors in each spatial direction, needed for the computation of the fluxes of leaves; if these nearest neighbors do not exist, we create them as virtual leaves. Brothers are also considered nearest cousins. The leaves of the tree are the control volumes from which we form the adaptive mesh. Here, the property of the tree being graded means that grid refinement and coarsening is governed by that two neighboring control volumes cannot differ by

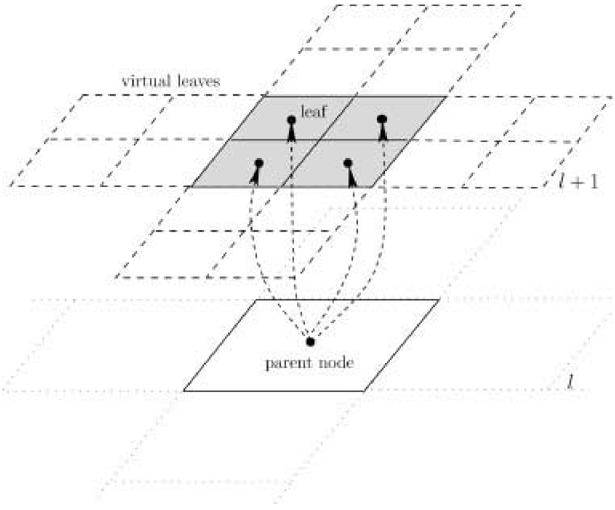

FIG. 1. A graded tree of two resolution levels. On the fine level: leaves (gray) and virtual leaves (dashed lines). On the coarse level: a parent node with its neighbors (dotted lines).

more than one level in the tree. This is equivalent to the concept of the "one-irregular" rule (see e.g., [35]). We denote by Λ the set of all nodes of the tree and by $\mathcal{L}(\Lambda)$ the restriction of Λ to the leaves. We apply this MR representation to the spatial part of the function $\mathbf{u} = (v, u_{\mathrm{e}}, w)$, which corresponds to the numerical solution of the underlying problem for each time step, so we need to update the tree structure for the proper representation of the solution during the evolution. To this end, we apply a thresholding strategy, but always keep the graded tree structure of the data. Once the thresholding is performed, we add to the tree a safety zone, so the new tree may contain the adaptive mesh for the next time step. The safety zone is generated by adding one finer level to the tree in all possible nodes without violating the graded tree data structure. This device, first proposed by Harten [22], ensures that the graded tree adequately represents the solution in the next time step, and its effectiveness depends strongly on the assumption of finite propagation speed of sharp fronts.

Let us now suppose that the tree has only two levels l and $l+1$. (The reasoning based on this assumption is straightforwardly extensible to an arbitrarily larger tree.) To ensure conservativity of the scheme, we compute only the fluxes at level $l+1$ and we set the ingoing flux on the leaf at level l equal to the sum of the outgoing fluxes on the leaves of level $l+1$ sharing the same edge

$$F_{(i+1,j),l\to(i,j),l} = F_{(2i+1,2j),l+1\to(2i+2,2j),l+1} + F_{(2i+1,2j+1),l+1\to(2i+2,2j+1),l+1}. \tag{4.5}$$

It is known that this choice decreases the number of costly flux evaluations without loosing the conservativity in the flux computation, and this represents a real advantage when using a graded tree structure, see e.g., [9] for more details. This advantage is lost for a non-graded tree structure, for which fluxes for leaves on an immediately finer level are not always available.

The data compression rate [11,12] $\eta := \mathcal{N}/(2^{-2L}\mathcal{N} + \#\mathcal{L}(\Lambda))$ measures the improvement in data compression. Here, $\mathcal{N}$ is the number of control volumes in the full finest grid at level L, and $\#\mathcal{L}(\Lambda)$ is the number of leaves. The speed-up $\mathcal{V}$ between the CPU time of the numerical solution obtained by the FV method and the CPU time of the numerical solution obtained by the MR method is defined by $\mathcal{V} := \text{CPU time}_{\text{FV}}/\text{CPU time}_{\text{MR}}$.

D. Selection of the Threshold Parameter

Error Analysis for Conservation Laws and Strictly Parabolic Equations. Let us recall the main results of the error analysis conducted in [8] and [9] for scalar, one-dimensional conservation laws and strictly parabolic equations, respectively. These results rely on certain properties of the respective (explicit) reference FV schemes, such as the CFL stability condition, the contraction property in L^1 norm and an order of convergence. We decompose the global error between the cell average values of the exact solution vector at the level L, denoted here by u_{ex}^L, and those of the MR computation with a maximal level L, denoted by u_{MR}^L, into two errors:

$$\left\| u_{\text{ex}}^L - u_{\text{MR}}^L \right\| \leq \left\| u_{\text{ex}}^L - u_{\text{FV}}^L \right\| + \left\| u_{\text{FV}}^L - u_{\text{MR}}^L \right\|. \tag{4.6}$$

The first term on the right-hand side, called discretization error, refers to the FV scheme on a uniform grid at the finest level L. For both a scalar, one-dimensional conservation law and a strictly parabolic equation, the order of convergence, denoted by $\tilde{\alpha}$, of the corresponding reference FV scheme is known ($\tilde{\alpha} = 1/2$ and $\tilde{\alpha} = 2$, respectively), which permits to obtain the estimate

$$\left\| u_{\text{ex}}^L - u_{\text{FV}}^L \right\| \leq C_1 2^{-\tilde{\alpha} L} \tag{4.7}$$

for a constant $C_1 > 0$. For the second term in the right-hand side of (4.6), called perturbation error, Cohen et al. [8] assume that the details on a level l are deleted when they are smaller than a prescribed tolerance ε_l. Then they show that if the discrete time evolution operator is contractive in the chosen norm, and if ε_l is given by (4.4), then the perturbation error accumulates in time and satisfies $\|u_{\text{FV}}^L - u_{\text{MR}}^L\| \leq C_2 n \varepsilon$, where $C_2 > 0$ and n denotes the number of time steps. At a fixed time $T = n\Delta t$, this gives

$$\left\| u_{\text{FV}}^L - u_{\text{MR}}^L \right\| \leq C_2 \frac{T}{\Delta t} \varepsilon, \quad C_2 > 0. \tag{4.8}$$

For the equations considered in [8, 9], the reference FV scheme converges under the CFL-type condition

$$\Delta t \leq \frac{h^2}{ph + q}, \tag{4.9}$$

where the constants p and q depend on the coefficients of the equation under consideration, and h is the meshwidth of the finest grid, i.e., $h = \tilde{C}(\Omega)2^{-L}$, where $\tilde{C}$ depends on the dimension and shape of the computational domain. Consequently, if Δt denotes the largest time step possible, then this quantity can be expressed in terms of 2^{-L} if we consider equality in (4.9), i.e.,

$$\Delta t = \frac{[\tilde{C}(\Omega)]^2 2^{-2L}}{\tilde{C}(\Omega) 2^{-L} p + q}. \tag{4.10}$$

It is desirable to choose the level-dependent thresholds in such a way that the total error, i.e., the error between the exact solution and the adaptive solution that is projected to the reference fine mesh, remains of the same order as the discretization error. For this purpose, one has to balance the discretization error and the perturbation error, which means that these two errors should be of the same order as h, or equivalently L varies. To derive an expression for ε from this requirement, we observe that the right-hand sides of (4.7) and (4.8) must be proportional, or equivalently,

$$\varepsilon \propto 2^{-\tilde{\alpha} L} \Delta t. \tag{4.11}$$

In light of (4.9), this yields a proportionality of the type

$$\varepsilon = \hat{C}2^{-\tilde{\alpha}L}\frac{[\tilde{C}(\Omega)]^2 2^{-2L}}{\tilde{C}(\Omega)2^{-L}p+q}, \tag{4.12}$$

from which one may deduce a value of the reference tolerance $\varepsilon_{\mathrm{R}} = \varepsilon$, provided that the factor of proportionality $\hat{C}$ can be determined, for example from suitable experiments, as is done in [9].

Reference Tolerance for the Bidomain System. The previous derivation of the reference tolerance is supported by a rigorous analysis only in those cases where the order of convergence $\tilde{\alpha}$ is known, and the discrete time evolution operator is contractive in the chosen norm. These properties hold for a scalar, one-dimensional first-order conservation law and a second-order parabolic equation studied in the references given above. For finite volume discretizations of the bidomain and monodomain models, however, an exact rate of convergence has not yet been derived. Nevertheless, in [10–12] we demonstrate that also for degenerate parabolic equations and reaction-diffusion systems, an equation of the type (4.12) may be employed to determine ε_{R}. We apply the same methodology to determine a reference tolerance for the problem at hand. On the basis of preliminary numerical experiments (obtained in a similar fashion as in [11]), for our examples, we obtained the approximate value $\tilde{\alpha} = 1.09$, which stands for the experimental accuracy order of the FV reference scheme. Using this, the global CFL condition of the reference FV scheme (3.6), the fact that $h = |\Omega|^{1/2}2^{-L}$, and the requirement that the proportionality (4.11) should hold, we obtain the following analog of (4.10):

$$\begin{aligned}\varepsilon_{\mathrm{R}} = C2^{-(\tilde{\alpha}+2)L}\biggl(&|\Omega| \max_{(i,j,l)\in\mathcal{L}(\Lambda)}\Bigl(\bigl|I_{\mathrm{ion},(i,j)}\bigr| + 2\bigl|I_{\mathrm{app},(i,j)}\bigr|\Bigr)\\ &+ |\Omega|^{3/2}2^{2+L}\max_{(i,j,l)\in\mathcal{L}(\Lambda)}\Bigl(\bigl|\mathbf{M}_{\mathrm{i},(i,j,l)}\bigr| + \bigl|\mathbf{M}_{\mathrm{e},(i,j,l)}\bigr|\Bigr)\biggr)^{-1},\end{aligned} \tag{4.13}$$

which will give the so-called reference tolerance to be used for the numerical examples in Section VI. To determine an acceptable value for the factor C, a series of computations with different tolerances are needed in each case, prior to final computations. Essentially, we select the largest available candidate value for C such that the same order of accuracy (same slopes for the error computation) as that of the reference FV scheme is maintained. In [7], the authors prove for scalar, one-dimensional nonlinear conservation laws, that the threshold error is stable in the sense that the constant C is uniformly bounded and, in particular, does not depend on the threshold value ε_{R}, the number of refinement levels L and the number of time steps n. In our case, even when a rigorous proof is still missing for the system considered in the present work, from the previous deduction and our numerical experiments (see Fig. 2 in Section VI) we see a similar behaviour for C. As in previous works [8, 10–12], here the reference tolerance ε_{R} remains fixed for all times. It is certainly possible to recompute ε_{R} at each time step, but this will usually mean that one has to perform additional computations to determine the value of C, and it seems unlikely that this procedure makes the scheme more efficient.

To measure errors between a reference solution u and an approximate solution u_{MR} obtained using MR, we will use L^p-errors: $e_p = \|u^n - u^n_{\mathrm{MR}}\|_p$, $p = 1, 2, \infty$, where

$$e_\infty = \max_{1\le i,j\le N}\bigl|u^n_{i,j} - u^n_{\mathrm{MR}i,j,L}\bigr|; \quad e_p = \left(\frac{1}{N^2}\sum_{i,j=1}^{N}\bigl|u^n_{i,j} - u^n_{\mathrm{MR}i,j,L}\bigr|^p\right)^{1/p}, \quad p = 1, 2.$$

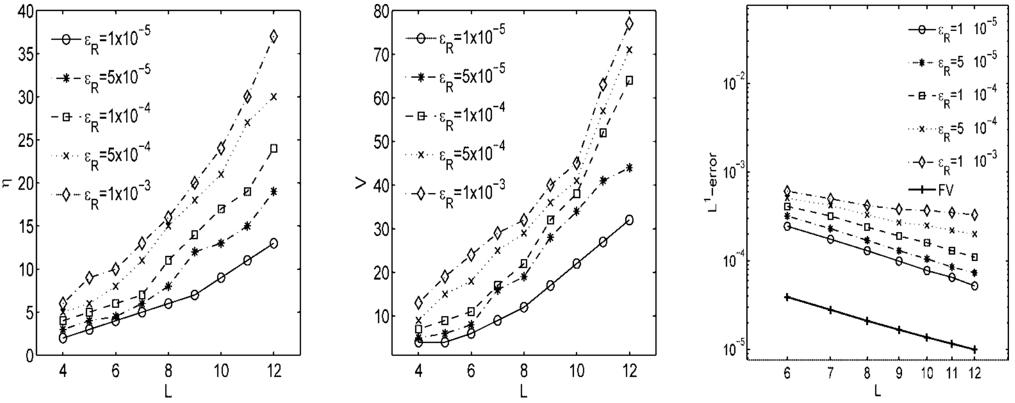

FIG. 2. Example 2 (bidomain model): data compression rate η (left), speed-up factor $\mathcal{V}$ (middle) and L^1-errors for different levels L and values of ε_{R} (right), at $t = 2.0\,\mathrm{ms}$.

where $u^n_{\mathrm{MR}i,j,L}$ is the value on the finest level L obtained by prediction from the corresponding leaf, and N is the number of control volumes in the finest mesh at level L in the x and $y-$direction, i.e., $N^2 = \mathcal{N}$.

V. TIME-STEP ACCELERATING METHODS

A. Local Time Stepping

We use a version of the locally varying time stepping strategy introduced by Müller and Stiriba [26] and summarize here its principles. The basic idea is to enforce a local CFL condition by using the same CFL number for all levels, and evolving all leaves on level l using the local time step size

$$\Delta t_l = 2^{L-l}\Delta t, \quad l = L-1,\ldots,0,$$

where $\Delta t = \Delta t_L$ corresponds to the time step size on the finest level L. This strategy allows to increase the time step for the major part of the adaptive mesh without violating the CFL stability condition. The synchronization of the time stepping for the portions of the solution lying on different resolution levels will be automatically achieved after 2^l time steps using Δt_l. To additionally save computational effort, the tree is updated only each odd intermediate time step $1, 3, \ldots, 2^L - 1$, and furthermore, the projection and prediction operators are performed only on levels occupied by the leaves of the current tree. For the rest of the intermediate time steps, we use the current (old) tree structure. For the sake of synchronization and conservativity of the flux computation, for coarse levels (levels without leaves), we use the same diffusive fluxes and sources computed in the previous intermediate time step, because the cell averages on these levels are the same as in the previous intermediate time step. Only for levels containing leaves, we compute fluxes in the following way: if there is a leaf at the corresponding edge and at the same resolution level l, we simply perform a flux computation using the brother leaves, and the virtual leaves at the same level if necessary; and if there is a leaf at the corresponding cell edge but on a finer resolution level $l + 1$ (i.e., this edge is an interface edge), the flux will be determined as in (4.5), i.e., we compute the fluxes at a level $l + 1$ on the same edge, and we set the ingoing flux on the corresponding edge at level l equal to the sum of the outgoing fluxes on the sons cells of level $l+1$ (for the same edge). We recall that the graded tree structure ensures that two neighboring control volumes of the adaptive mesh do not differ by more than one resolution level, which is equivalent to the satisfaction of the one-irregular rule.

To always have at hand the computed fluxes as in (4.5), we need to perform the locally varying time stepping recursively from fine to coarse levels.

B. A Runge-Kutta-Fehlberg Method

To upgrade the FV scheme described in Section III to at least second order so that second-order accuracy both in space and time is effective, we use an RKF method, which, apart from providing the necessary accuracy, also allows an adaptive control of the time step. For our models, we consider a vector-valued RKF method, i.e., $\mathbf{u} = (v, u_{\mathrm{e}}, w)$ and its time- discretized form at step m, denoted by $\mathbf{u}^m$. For ease of discussion, we assume that the problem is written as $\partial_t \mathbf{u} = \mathcal{A}(t, \mathbf{u})$.

We use two Runge-Kutta methods of orders $p = 3$ and $p - 1 = 2$

$$\hat{\mathbf{u}}^{m+1} = \mathbf{u}^m + \hat{b}_1\bar{\kappa}_1 + \hat{b}_2\bar{\kappa}_2 + \hat{b}_3\bar{\kappa}_3, \quad \check{\mathbf{u}}^{m+1} = \mathbf{u}^m + \check{b}_1\bar{\kappa}_1 + \check{b}_2\bar{\kappa}_2 + \check{b}_3\bar{\kappa}_3,$$

where

$$\begin{aligned} &\bar{\kappa}_1 := \Delta t\mathcal{A}(t^m, \mathbf{u}^m), \quad \bar{\kappa}_2 := \Delta t\mathcal{A}(t^m + c_2\Delta t, \mathbf{u}^m + a_{21}\bar{\kappa}_1), \\ &\bar{\kappa}_3 := \Delta t\mathcal{A}(t^m + c_3\Delta t, \mathbf{u}^m + a_{31}\bar{\kappa}_1 + a_{32}\bar{\kappa}_2), \end{aligned} \tag{5.1}$$

and the coefficients of the RK3(2) method are $c_2 = a_{21} = 1, c_3 = \frac{1}{2}, a_{31} = a_{32} = \frac{1}{4}, \hat{b}_1 = \hat{b}_2 = \frac{1}{6}$, $\hat{b}_3 = \frac{2}{3}$, $\check{b}_1 = \check{b}_2 = \frac{1}{2}$, and $\check{b}_3 = 0$. These values yield an optimal pair of embedded TVD-RK methods of orders two and three. The truncation error between the two approximations for $\mathbf{u}^{m+1}$ is estimated by

$$\bar{\delta}_{\mathrm{old}} := \hat{\mathbf{u}}^{m+1} - \check{\mathbf{u}}^{m+1} = \sum_{i=1}^{p}(\hat{b}_i - \check{b}_i)\bar{\kappa}_i \sim (\Delta t)^p, \quad \delta_{\mathrm{old}} := \|\bar{\delta}_{\mathrm{old}}\|_\infty. \tag{5.2}$$

Then we can adjust the step size to achieve a prescribed accuracy $\delta_{\mathrm{desired}}$ in time. The new time step is determined by $\Delta t_{\mathrm{new}} = \Delta t_{\mathrm{old}}|\delta_{\mathrm{desired}}/\delta_{\mathrm{old}}|^{1/p}$ with $p = 3$. To avoid excessively large time steps, we use a limiter function $\mathcal{S}(t) := (\mathcal{S}_0 - \mathcal{S}_{\min})\exp(-t/\Delta t) + \mathcal{S}_{\min}$, where we choose $\mathcal{S}_0 = 0.1$ and $\mathcal{S}_{\min} = 0.01$. The new time step Δt_{new} is then defined as

$$\Delta t_{\mathrm{new}} = \begin{cases} \Delta t_{\mathrm{old}}|\delta_{\mathrm{desired}}/\delta_{\mathrm{old}}|^{1/p} & \text{if } |(\Delta t_{\mathrm{new}} - \Delta t_{\mathrm{old}})/\Delta t_{\mathrm{old}}| \le \frac{1}{2}\mathcal{S}(t, \Delta t_{\mathrm{old}}), \\ \frac{1}{2}\mathcal{S}(t, \Delta t_{\mathrm{old}})\Delta t_{\mathrm{old}} + \Delta t_{\mathrm{old}} & \text{otherwise.} \end{cases} \tag{5.3}$$

Notice that Δt_{new} is the time step size for computing $\mathbf{u}^{m+2}$. More details on the RKF scheme and its implementation can be found in [7, 11].

VI. NUMERICAL EXAMPLES

We present three test cases showing the efficiency of the previously described methods in capturing the dynamical evolution of electro-physiological waves for both the monodomain and bidomain models. We are dealing with multicomponent solutions, but use a single mesh to represent the

TABLE I. Parameters chosen in the simulations.

	Example 1	Example 2	Example 3
	(monodomain model)	(bidomain model)	(bidomain model)
Kinetics	FitzHugh-Nagumo	Mitchell-Schaeffer	Mitchell-Schaeffer
		$\eta_1 = 0.005$	$\eta_1 = 0.005$
	$a = 0.16875$	$\eta_2 = 0.1$	$\eta_2 = 0.1$
	$b = 1.0$	$\eta_3 = 1.5$	$\eta_3 = 1.5$
	$\lambda = -100$	$\eta_4 = 7.5$	$\eta_4 = 7.5$
	$\theta = 0.25$	$\eta_5 = 0.1$	$\eta_5 = 0.1$
		$v_p = 100\,\mathrm{mV}$	$v_p = 100\,\mathrm{mV}$
Model parameters			
$\beta\,[\mathrm{cm}^{-1}]$	1.0	2000	2000
γ	0.01	–	–
Ω	$[0, 1\,\mathrm{cm}]^2$	$[0, 0.5\,\mathrm{cm}]^2$	$[0, 0.5\,\mathrm{cm}]^2$
$c_\mathrm{m}\,[\mathrm{mF/cm}^2]$	1.0	1.0	1.0
$\sigma_\mathrm{i}^\mathrm{l}\,[\mathrm{Ohm}^{-1}\mathrm{cm}^{-1}]$	–	6.0	6.0
$\sigma_\mathrm{i}^\mathrm{t}\,[\mathrm{Ohm}^{-1}\mathrm{cm}^{-1}]$	–	0.6	0.6
$\sigma_\mathrm{e}^\mathrm{l}\,[\mathrm{Ohm}^{-1}\mathrm{cm}^{-1}]$	–	24.0	24.0
$\sigma_\mathrm{e}^\mathrm{t}\,[\mathrm{Ohm}^{-1}\mathrm{cm}^{-1}]$	–	12.0	12.0
$R_\mathrm{m}\,[\mathrm{Ohm\,cm}^2]$	–	2.0×10^4	2.0×10^4
MR parameters			
L	10	9	9
$\mathcal{N}$	262144	65536	65536
r	3	5	5
ε_R	1.0×10^{-3}	5.0×10^{-4}	2.5×10^{-3}
RKF/LTS parameters			
δ_desired	–	1.0×10^{-4}	1.0×10^{-3}
$\mathcal{S}_0$		0.1	0.1
$\mathcal{S}_\mathrm{min}$	–	0.01	0.01
CFL_l	–	$\mathrm{CFL}_{l=0} = \mathrm{CFL}_{t=0} = 0.5$, $\mathrm{CFL}_l = 2^l \mathrm{CFL}_{l=0}$	–

The model parameters have been chosen according to [18].

vector of relevant variables. In the bidomain model, the anisotropies, mesh structures, and the size of the problem cause the sparse linear system corresponding to (3.4) to be ill-conditioned. The system matrix is symmetric, positive definite and sparse. The system is solved in each time step, which is done directly by a Cholesky factorization. All the parameters used in our computations are listed in Table I.

A. Example 1

In this example, we consider the monodomain model (2.8) with homogeneous Neumann boundary conditions. We consider in (2.8) $(1 + \mu)^{-1}\mathbf{M}_\mathrm{i} := \mathrm{diag}(\gamma, \gamma)$ and the initial data

$$v^0(x, y) = \left(1 - \frac{1}{1 + \exp(-50(x^2 + y^2)^{1/2} - 0.1)}\right)\,\mathrm{mV}, \quad w_0 = 0\,\mathrm{mV}.$$

After 4 ms, an instantaneous stimulus is applied in $(x_0, y_0) = (0.5\,\mathrm{cm}, 0.5\,\mathrm{cm})$ to the membrane potential v

$$\frac{\mu}{1 + \mu} I_\mathrm{app} := \begin{cases} 1\,\mathrm{mV} & \text{if } (x - x_0)^2 + (y - y_0)^2 < 0.04\,\mathrm{cm}^2, \\ 0\,\mathrm{mV} & \text{otherwise.} \end{cases}$$

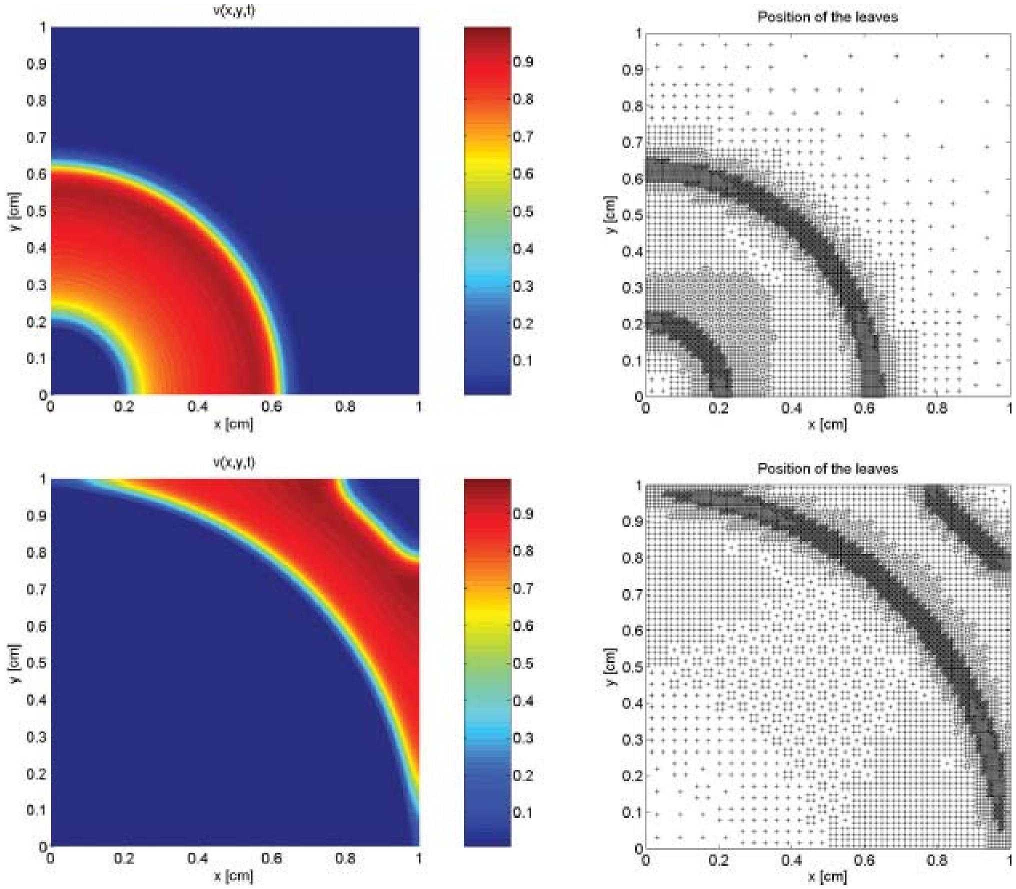

FIG. 3. Example 1 (monodomain model): Numerical solution for v (measured in mV) (left) and leaves of the corresponding tree (right) at $t = 1.5$ ms (top) and $t = 3.5$ ms (bottom). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

We compute normalized errors by comparison with a reference solution obtained with a fine mesh calculation with $1024^2 = 1{,}048{,}576$ control volumes. The time evolution is made using a first-order explicit Euler scheme. Plots of the numerical solution with the corresponding adaptively refined meshes at different times are shown in Figures 3 and 4.

Table II shows that the normalized errors are controlled to be of the same order of ε_{R}. Moreover, the MR algorithm is effective, since we have high rates of memory compression and speed-up.

B. Example 2

In Examples 2 and 3, we present numerical results for the bidomain model where the fibers form an angle of $\pi/4$ with the x-axis.

In Example 2, the initial datum is given by a stimulus applied on the extracellular potential u_{e} in the center of the domain, while both v and the gating variable w are initially set to zero (see Fig. 5). The units for v, u_e and w are mV.

We show in Figures 6 and 7 a sequence of snapshots after an initial stimulus applied to the center of the domain, corresponding to transmembrane potential v, extracellular potential u_{e} and

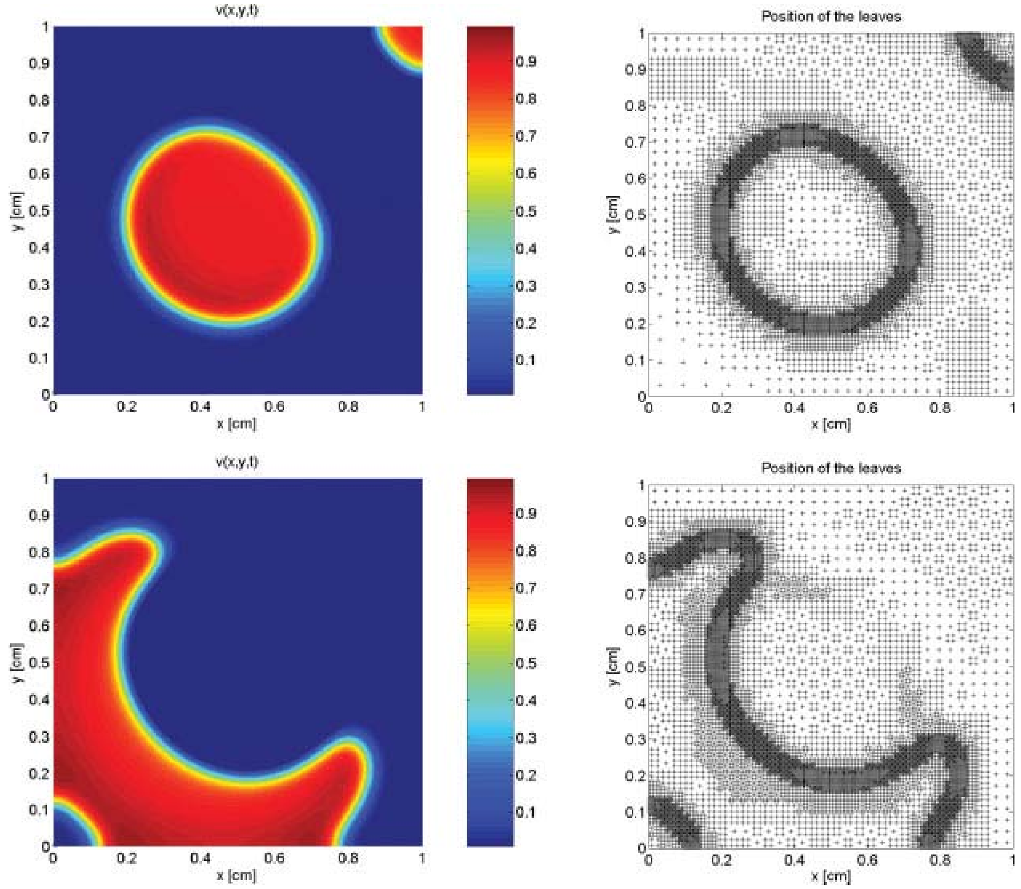

FIG. 4. Example 1 (monodomain model): Numerical solution for v (measured in mV) (left) and leaves of the corresponding tree (right) at $t = 4.5\,\mathrm{ms}$ (top) and $t = 5.5\,\mathrm{ms}$ (bottom). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

adaptive mesh. The width of the fronts exhibited can be clearly seen in the one-dimensional plot of the section $y = 2.5$ of the solution for u_e given in Figure 8.

Table III illustrates the efficiency and accuracy of the base MR method in terms of CPU ratio $\mathcal{V}$, the compression rate η, and normalized errors. By using MR, we obtain an average data compression rate of 17 and an increasing speed-up rate up to 26.09. Moreover, the errors in three different

TABLE II. Example 1 (monodomain model): Corresponding simulated time, CPU ratio $\mathcal{V}$, compression rate η and normalized errors for v, using an MR method.

Time t (ms)	$\mathcal{V}$	η	L^1−error	L^2−error	L^∞−error
0.0		170.22	3.99×10^{-4}	2.47×10^{-4}	4.31×10^{-4}
1.5	27.81	37.56	4.63×10^{-4}	1.96×10^{-4}	4.97×10^{-4}
3.5	26.47	29.89	4.82×10^{-4}	4.05×10^{-4}	5.23×10^{-4}
4.5	31.41	28.12	5.31×10^{-4}	4.29×10^{-4}	7.48×10^{-4}
5.5	30.62	24.70	6.79×10^{-4}	6.20×10^{-4}	1.04×10^{-3}

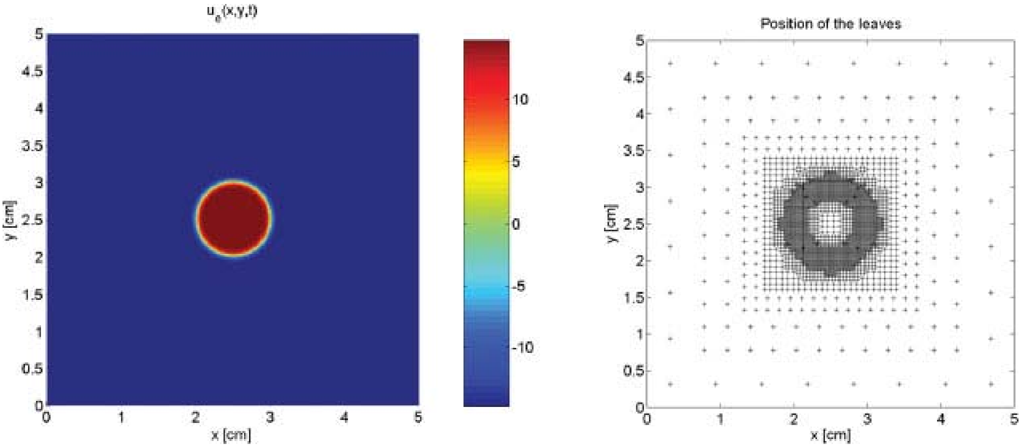

FIG. 5. Example 2 (bidomain model, one stimulus): Initial condition for the extracellular potential u_e (left) and leaves of the corresponding tree data structure (right). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

norms remain of the order of ε_R. Here we have computed normalized errors using a reference FV solution on a grid with $1024^2 = 1{,}048{,}576$ control volumes.

We select this example to compare the performances of the FV and MR methods with a global time step, the MR method with RKF adaptive global time stepping (MR-RKF), and the MR method with local time stepping (MR-LTS). The evolution of $\mathcal{V}$ and η for the MR versions and of the normalized L^1 and L^∞ errors for all these methods are displayed in Figure 9. These plots indicate that with RKF and LTS, η is of the same order during the time evolution, which means that the adaptive meshes for both methods should be roughly the same. Also, a substantial additional gain is obtained in the speed-up when comparing with an MR calculation using global time stepping: The MR-LTS method provides an additional speed-up factor of about 2, while with the RKF alternative we obtain an additional speed-up of about 4. This effect could be explained in part by the lack of necessity of a synchronization procedure for the RKF computations, and the fact that the CFL condition (3.6) is not imposed during the time evolution with the MR-RKF method, allowing larger time steps. (Although condition (3.6) guarantees numerical stability of the solutions, in practice this is observed to be a fairly conservative estimate, and moderately larger time steps may be used.) We also conclude that the errors of the MR-LTS computations are kept of the same order as the errors obtained with global time stepping, while the errors incurred by using the MR-RKF method are larger during the whole time evolution. A potential explanation of this observation seems to be that while the η versus t plot of Figure 9 indicates that the spatial representation of the solution with the three MR variants seems to be roughly the same, the V versus t plot shows that MR with RKF is significantly faster than the two other variants, i.e., our MR-RKF strategy allows for larger time steps, even though within the MR-RKF framework the solution is evolved globally with the same, potentially small time step. Consequently, the MR-RKF method produces larger errors here than the MR-LTS method, as is visible from the L^1/L^∞-error versus t plots of Figure 9. It seems, however, that this phenomenon is basically due to our adhoc choice of the RKF parameters δ_{desired}, $\mathcal{S}_0$, and $\mathcal{S}_{\min}$.

Finally, we select this example for a brief discussion of the CPU time required for one iteration. Of course, this quantity will strongly depend on the particular time instant considered. Basically, if

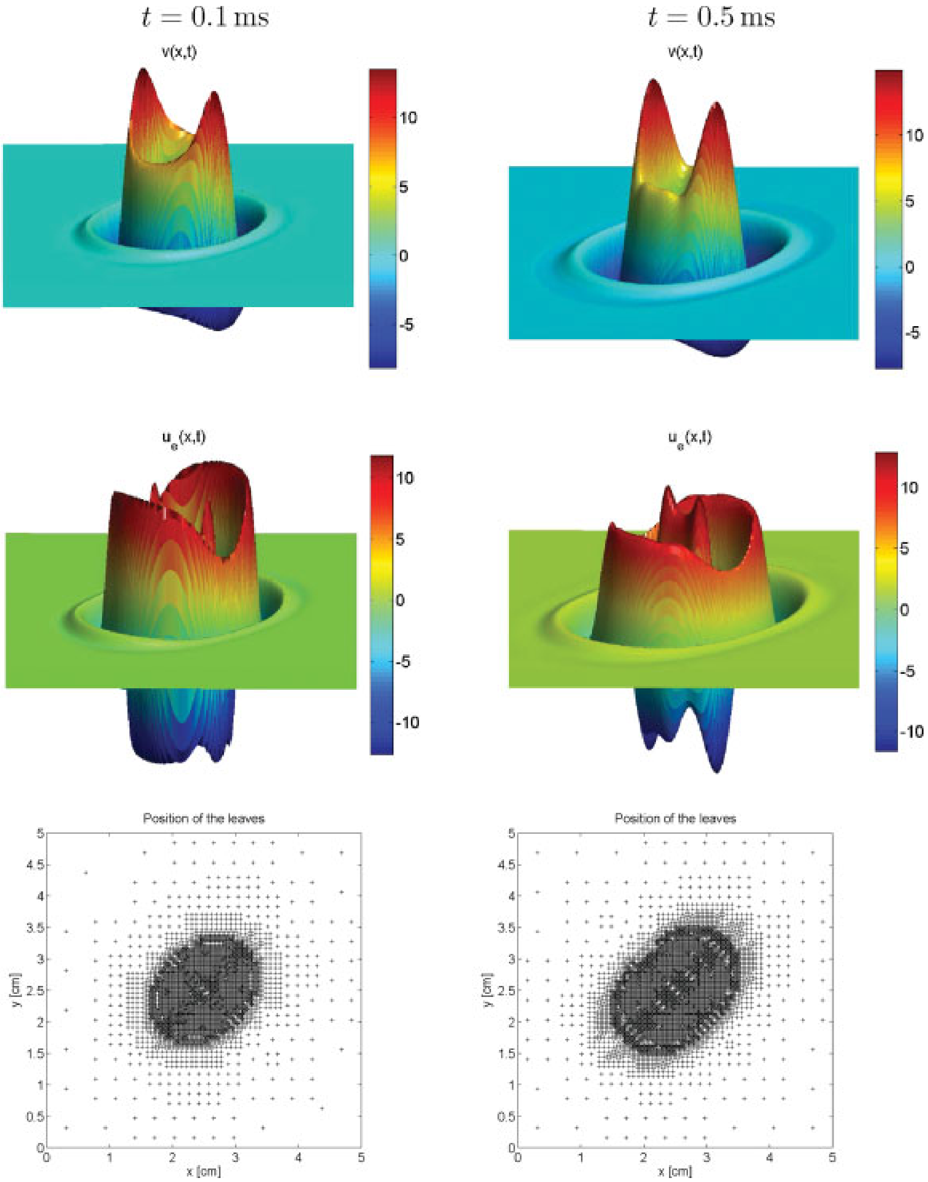

FIG. 6. Example 2 (bidomain model, one stimulus): Numerical solution for v (top) and u_e (middle) (in mV), and leaves of the corresponding tree data structure (bottom) at $t = 0.1$ ms (left) and $t = 0.5$ ms (right). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

high levels of resolution are needed in a considerable part of the domain, then the most expensive step of the algorithm (which is always the flux computation) will represent a larger percentage of CPU time. For Example 2, at $t = 5.0$ ms, we have the following distribution of CPU time consumption (we include initial and final steps, not needed for every iteration): initialization of parameters and creation of the initial tree structure: 15%; determination of the leaves and sets

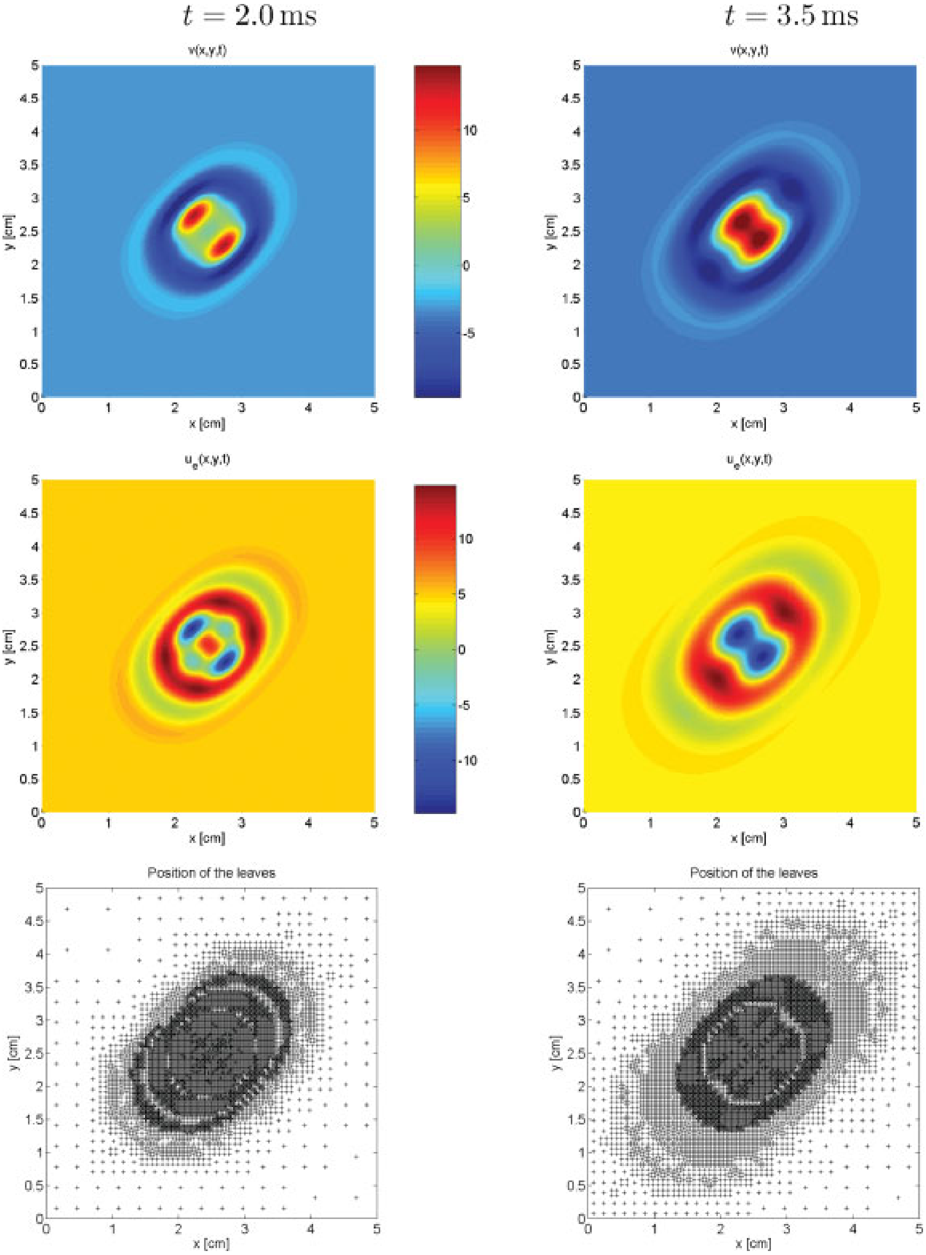

FIG. 7. Example 2 (bidomain model, one stimulus): Numerical solution for v (top) and u_e (middle) (in mV), and leaves of the corresponding tree data structure (bottom) at $t = 2.0$ ms (left) and $t = 3.5$ ms (right). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

of virtual leaves: 10%; computation of the discretized divergence operator for all leaves: 30%; solution of a linear system (the one which is solved via Cholesky factorization): 20%; updating the tree structure: 15%; saving meshes, leaves and cell averages; and rest of the computations: 10%. Overall, arithmetic operations related to the actual computation of the numerical solution

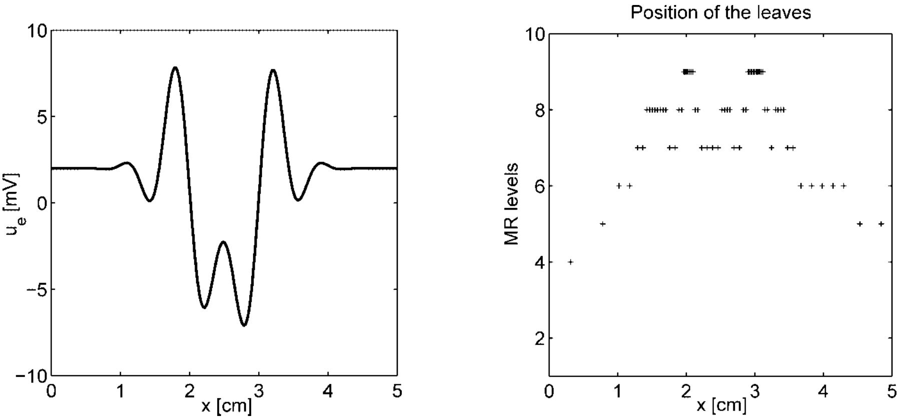

FIG. 8. Example 2 (bidomain model, one stimulus): Profile of the numerical solution for u_{e} at $y = 2.5$ (left), and leaves of the corresponding tree data structure (right) at $t = 5.0\,\mathrm{ms}$.

consume roughly 50% of CPU time; the remainder goes into the "overhead" of the administration of the multiresolution representation and the graded tree.

C. Example 3

We now consider an initial stimulus at the center of Ω, later at $t = 0.2\,\mathrm{ms}$, we apply another instantaneous stimulus to the northwest corner of Ω, and at $t = 1.0\,\mathrm{ms}$ we apply a third stimulus of the same magnitude to the northeast and southwest corners. The system is evolved and we show snapshots of the numerical solution for v and u_{e} and the adaptive mesh. The numerical results shown in Figures 10 and 11 clearly indicate the anisotropic orientation of the fibers.

VII. CONCLUSIONS

We address the application of a MR method for FV schemes combined with LTS and RKF adaptive time stepping for solving the bidomain equations. The numerical experiments illustrate that these methods are efficient and accurate enough to simulate the electrical activity in myocardial tissue with affordable effort. This is a real advantage in comparison with more involved methods

TABLE III. Example 2 (bidomain model, one stimulus): Corresponding simulated time, CPU ratio $\mathcal{V}$, compression rate η, and normalized errors.

Time t (ms)	$\mathcal{V}$	η	Potential	L^1−error	L^2−error	L^∞−error
0.1	13.74	19.39	v	3.68×10^{-4}	8.79×10^{-5}	6.51×10^{-4}
			u_{e}	2.01×10^{-4}	6.54×10^{-5}	5.22×10^{-4}
0.5	21.40	17.63	v	4.06×10^{-4}	9.26×10^{-5}	6.83×10^{-4}
			u_{e}	2.79×10^{-4}	8.72×10^{-5}	5.49×10^{-4}
2.0	25.23	17.74	v	4.37×10^{-4}	1.25×10^{-4}	6.88×10^{-4}
			u_{e}	3.48×10^{-4}	9.44×10^{-5}	6.11×10^{-4}
5.0	26.09	16.35	v	5.29×10^{-4}	1.94×10^{-4}	7.20×10^{-4}
			u_{e}	4.15×10^{-4}	1.06×10^{-4}	6.32×10^{-4}

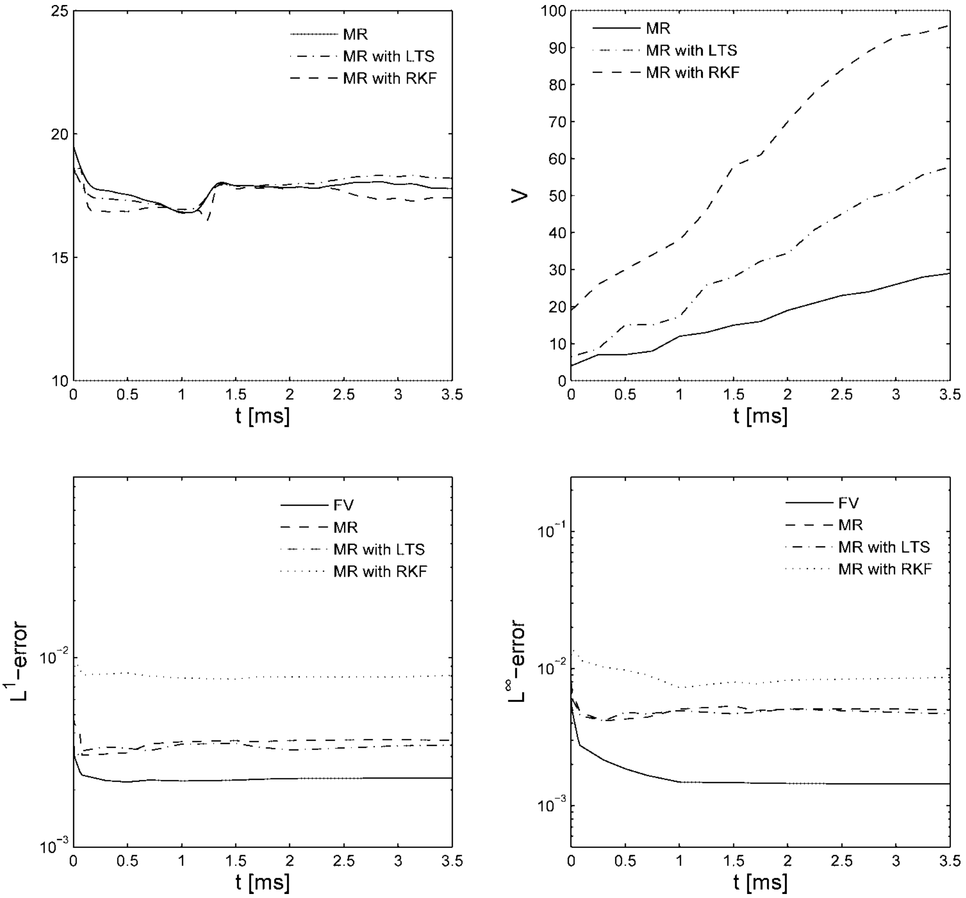

FIG. 9. Example 2 (bidomain model, one stimulus): Time evolution for data compression rate η (top left), speed-up rate $\mathcal{V}$ (top right), and normalized L^1 errors (bottom left) and L^∞ errors (bottom right) for different methods: MR scheme with global time step, MR with locally varying time stepping and MR with RKF time stepping.

that require large scale computations on clusters. We here contribute to the recent work done by several groups in testing whether the combination of MR, LTS, and RKF strategies is indeed effective for a relevant class of problems.

Concerning the orders of the schemes involved, we mention that at least in the case of hyperbolic equations, it is known that when using MR schemes of high order, and even if the reference FV method is only of first order, the global scheme maintains the order of the MR reconstruction. In the case of the bidomain equations, we experimentally see the same behavior. This motivates our choice of the reference tolerance in such a way that the discretization and perturbation errors are of the same order (see Section IVD), and furthermore the order of the MR reconstruction is set to one, with $r = 3$ (as the order of the reference FV scheme), in Example 1. For Examples 2 and 3, we use a higher order reconstruction (of order 2, with $r = 5$), and this high order is inherited by the global method. This is why we choose an RKF method of order (3)2.

From a numerical point of view, the plateau-like structures, associated with very steep gradients, of typical solutions motivate the use of a locally refined adaptive mesh, since we require high

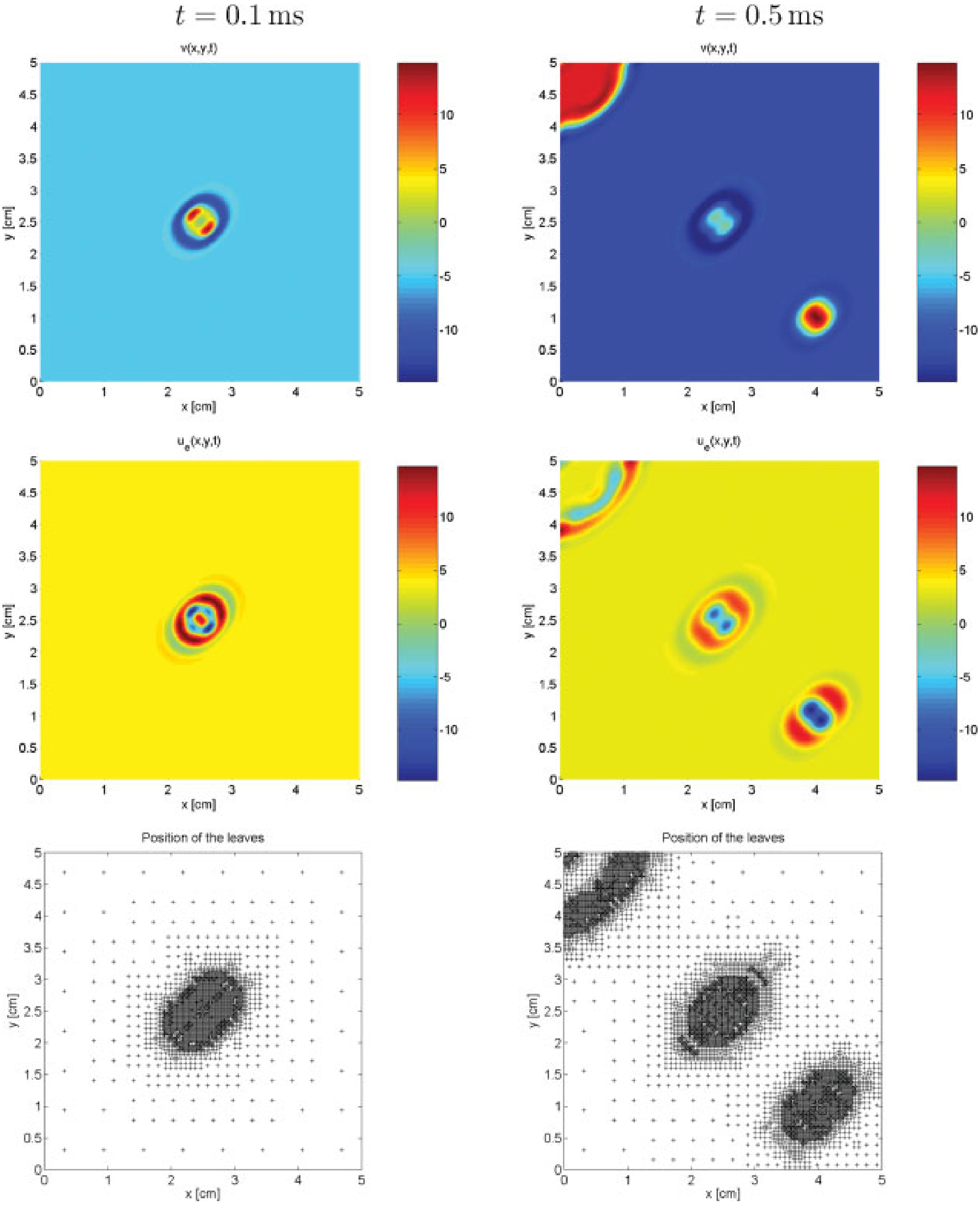

FIG. 10. Example 3 (bidomain model, three stimuli): Numerical solution for v (top) and u_e (middle) (in mV), and leaves of the corresponding tree data structure (bottom) at $t = 0.1$ ms (left) and $t = 0.5$ ms (right). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

resolution near these steep gradients only. These areas of strong variation occupy a very reduced part of the entire domain only, especially in the case of sharp fronts. Consequently, our gain will be less significant in the presence of chaotic electrical activity or when multiple waves interact in the considered tissue.

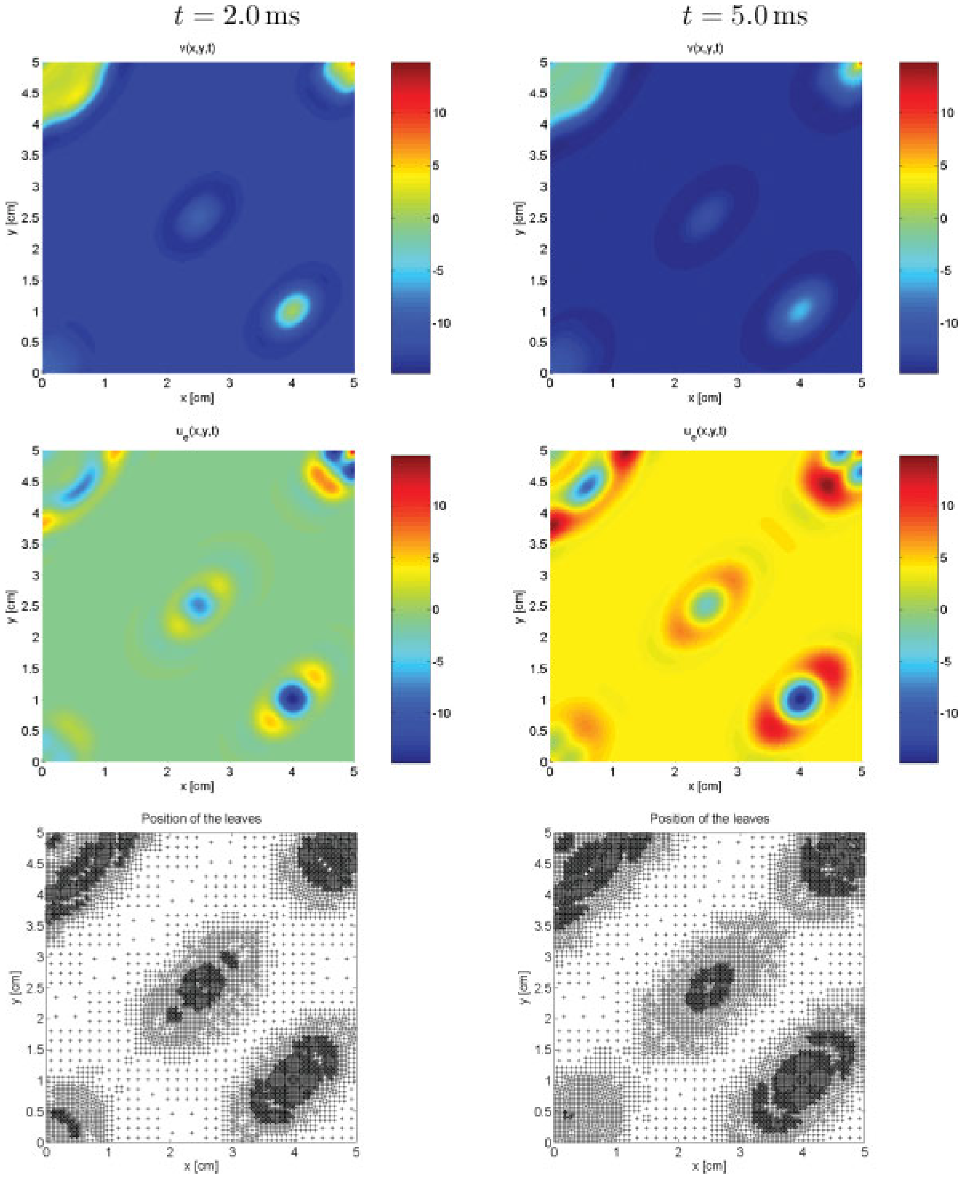

FIG. 11. Example 3 (bidomain model, three stimuli): Numerical solution for transmembrane potential v (top) and extracellular potential u_e (middle) (in mV), and leaves of the corresponding tree data structure (bottom) at $t = 2.0\,\mathrm{ms}$ (left) and $t = 5.0\,\mathrm{ms}$ (right). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

On the basis of our numerical examples, we conclude that using an LTS strategy, we obtain a substantial gain in CPU time speed-up of a factor of about 2 for finer levels while the errors between the MR-LTS solution and a reference solution are of the same order as those of the MR solution. On the other hand, using an MR-RKF strategy, we obtain an additional speed-up factor of about 4, but at the price of larger errors. However, in assessing our findings, it is important to recognize limitations. The high rates of compression obtained with our methods are problem-dependent and they may depend on the proper adjustment of parameters. We have only considered here very simple geometries, because all computations are concentrated on adaptivity and performance. Simulations on more complex and realistic geometries are part of possible future work.

Finally, we remark that the FV method given in Section III as well the MR framework detailed in Section IV are both straightforwardly extensible to the 3D case.

References

1. A. L. Hodgkin and A. F. Huxley, A quantitative description of membrane current and its application to conduction and excitation in nerve, J Physiol 117 (1952), 500–544.
2. J. Keener and J. Sneyd, Mathematical Physiology, Vols. I & II, 2nd Ed., Springer-Verlag, New York, 2009.
3. L. Tung, A bi-domain model for describing ischemic myocardial D-C currents, PhD thesis, MIT, Cambridge, MA, 1978.
4. M. Bendahmane and K. H. Karlsen, Analysis of a class of degenerate reaction-diffusion systems and the bidomain model of cardiac tissue, Netw Heterog Media 1 (2006), 185–218.
5. P. Colli Franzone, P. Deuflhard, B. Erdmann, J. Lang, and L. F. Pavarino, Adaptivity in space and time for reaction–diffusion systems in electro-cardiology, SIAM J Sci Comput 28 (2006), 942–962.
6. P. Colli Franzone and G. Savaré, Degenerate evolution systems modeling the cardiac electric field at micro- and macroscopic level, In: A. Lorenzi and B. Ruf, editors, Evolution equations, semigroups and functional analysis, Birkhäuser, Basel, 2002, pp. 49–78.
7. M. Domingues, S. Gomes, O. Roussel, and K. Schneider, An adaptive multiresolution scheme with local time-stepping for evolutionary PDEs, J Comput Phys 227 (2008), 3758–3780.
8. A. Cohen, S. Kaber, S. Müller, and M. Postel, Fully adaptive multiresolution finite volume schemes for conservation laws, Math Comp 72 (2001), 183–225.
9. O. Roussel, K. Schneider, A. Tsigulin, and H. Bockhorn, A conservative fully adaptive multiresolution algorithm for parabolic PDEs, J Comput Phys 188 (2003), 493–523.
10. M. Bendahmane, R. Bürger, R. Ruiz-Baier, and K. Schneider, Adaptive multiresolution schemes with local time stepping for two-dimensional degenerate reaction-diffusion systems, Appl Numer Math 59 (2009), 1668–1692.
11. R. Bürger, R. Ruiz, K. Schneider, and M. Sepúlveda, Fully adaptive multiresolution schemes for strongly degenerate parabolic equations in one space dimension, M2AN Math Model Numer Anal 42 (2008), 535–563.
12. R. Bürger, R. Ruiz, K. Schneider, and M. Sepúlveda, Fully adaptive multiresolution schemes for strongly degenerate parabolic equations with discontinuous flux, J Engrg Math 60 (2008), 365–385.
13. E. Cherry, H. Greenside, and C. S. Henriquez, Efficient simulation of three–dimensional anisotropic cardiac tissue using an adaptive mesh refinement method, Chaos 13 (2003), 853–865.
14. H. Chen and X.-H. Zhong, Global existence and blow-up for the solutions to nonlinear parabolic/elliptic system modelling chemotaxis, IMA J Appl Math 70 (2005), 221–240.
15. Y. Bourgault, Y. Coudière, and C. Pierre, Existence and uniqueness of the solution for the bidomain model used in cardiac electro-physiology, Nonlin Anal Real World Appl 10 (2009), 458–482.
16. W. Quan, S. Evans, and H. Hastings, Efficient integration of a realistic two-dimensional cardiac tissue model by domain decomposition, IEEE Trans Biomed Eng 45 (1998), 372–385.
17. J. Sundnes, G. T. Lines, and A. Tveito, An operator splitting method for solving the bidomain equations coupled to a volume conductor model for the torso, Math Biosci 194 (2005), 233–248.
18. K. Skouibine, N. Trayanova, and P. Moore, A numerically efficient model for simulation of defibrillation in an active bidomain sheet of myocardium, Math Biosci 166 (2000), 85–100.

19. M. J. Berger and J. Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, J Comput Phys 53 (1984), 482–512.

20. P. Colli Franzone and L. F. Pavarino, A parallel solver for reaction-diffusion systems in computational electro-cardiology, Math Models Meth Appl Sci 14 (2004), 883–911.

21. H. Saleheen and K. Ng, A new three–dimensional finite-difference bidomain formulation for inhomogeneous anisotropic cardiac tissues, IEEE Trans Biomed Eng 45 (1998), 15–25.

22. A. Harten, Multiresolution algorithms for the numerical solution of hyperbolic conservation laws, Comm Pure Appl Math 48 (1995), 1305–1342.

23. S. Müller, Adaptive Multiscale Schemes for Conservation Laws, Springer-Verlag, Berlin, 2003.

24. G. Chiavassa, R. Donat, and S. Müller, Multiresolution-based adaptive schemes for hyperbolic conservation, in: T. Plewa, T Linde, V. G. Weiss (Editors), Adaptive mesh refinement-theory and applications, Springer-Verlag, Berlin, 2003, pp. 137–159.

25. W. Dahmen, B. Gottschlich-Müller, and S. Müller, Multiresolution schemes for conservation laws, Numer Math 88 (2001), 399–443.

26. S. Müller and Y. Stiriba, Fully adaptive multiscale schemes for conservation laws employing locally varying time stepping, J Sci Comput 30 (2007), 493–531.

27. J. Sundnes, G. T. Lines, X. Cai, B. F. Nielsen, K.-A. Mardal, and A. Tveito, Computing the electrical activity in the heart, Springer-Verlag, Berlin, 2006.

28. C. Mitchell and D. Schaeffer, A two-current model for the dynamics of cardiac membrane, Bull Math Biol 65 (2001), 767–793.

29. P. Colli Franzone, L. F. Pavarino, and B. Taccardi, Simulating patterns of excitation, repolarization and action potential duration with cardiac bidomain and monodomain models, Math Biosci 197 (2005), 35–66.

30. P. R. Johnston, The effect of simplifying assumptions in the bidomain model of cardiac tissue: Application to ST segment shifts during partial ischaemia, Math Biosci 198 (2005), 97–118.

31. M. Bendahmane and K. H. Karlsen, Convergence of a finite volume scheme for the bidomain model of cardiac tissue, Appl Numer Math 59 (2009), 2266–2284.

32. Y. Coudière, C. Pierre, and R. Turpault, Solving the fully coupled heart and torso problems of electro cardiology with a 3D discrete duality finite volume method. HAL preprint (2006), available from `http://hal.archives-ouvertes.fr/ccsd-00016825`.

33. M. Bendahmane, R. Bürger, and R. Ruiz-Baier, A finite volume scheme for cardiao propagation in media with isotropic conductivites. Preprint, Universidad de Concepción; submitted.

34. M. Pennacchio and V. Simoncini, Efficient algebraic solution of reaction-diffusion systems for the cardiac excitation process, J Comput Appl Math 145 (2002), 49–70.

35. P. K. Moore, An adaptive finite element method for parabolic differential systems: some algorithmic considerations in solving in three space dimensions, SIAM J Sci Comput 21 (2000), 1567–1586.