Journal of Computational and Applied Mathematics 296 (2016) 550–563

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Operator splitting for the bidomain model revisited☆

Raymond J. Spiteri[a,*], Saeed Torabi Ziaratgahi[b]

[a] Department of Computer Science, University of Saskatchewan, Saskatoon, Saskatchewan, S7N 5C9, Canada

[b] Department of Mathematics and Statistics, University of Saskatchewan, Saskatoon, Saskatchewan, S7N 5E6, Canada

ARTICLE INFO

Article history:
Received 8 April 2015
Received in revised form 18 July 2015

Keywords:
Bidomain model
Operator splitting
Godunov method
Semi-implicit method

ABSTRACT

The bidomain model is a widely used mathematical model to describe the propagation of electricity in myocardial tissue. It consists of a multi-scale system of partial differential equations coupling the electrical activity at the tissue scale with that at the cellular scale. It is common to solve the bidomain model by using a separate numerical procedure for each scale. Two well-known, first-order time-integration methods for solving the bidomain model are the semi-implicit method of Southern et al. (2009) and the Godunov operator-splitting method (as described in Sundnes et al., 2006). Both methods decouple the numerical procedure at the cellular scale from that at the tissue scale but in slightly different ways. The methods are analysed in terms of their accuracy, and their relative performance is compared on one-, two-, and three-dimensional test cases. As suggested by the analysis, the test cases show that the Godunov method is significantly faster than the semi-implicit method for the same level of accuracy, specifically, between 5 and 15 times in the cases presented.

1. Introduction

In its report of May 2014, the World Health Organization identified ischaemic heart disease as the leading cause of death in the world in 2012 with 7.4 million deaths [1]. This represents a 23% increase over the number of deaths caused by ischaemic heart disease in 2000 and 13% of the total deaths caused by the top 10 causes of death in the world in 2012 [1]. The cost of heart disease is high, both in terms of human suffering and monetary cost, with more than one in four adults in the United States coping with some form of heart disease [2] and well over US$170 billion being spent annually on treatment [3]. Abnormalities in the electrical activity of the heart are often cited as the cause of many types of heart disease, e.g., stroke [4].

The bidomain model, introduced by Tung in 1978 [5], is one of the most widely used mathematical models to describe the electrical activity in myocardial tissue. It is a multi-scale model that couples the electrical activity at the cellular scale with the propagation of electricity at the tissue scale.

The bidomain model is challenging to solve in realistic situations. A human heart has approximately 10 billion muscle cells, and there are typically tens of state variables per cell. Although the individual simulation of the state of each cell is not necessary to obtain clinically useful data, the computational requirements to perform a realistic simulation of even a single heartbeat are nonetheless formidable. Consequently, a great deal of effort has been devoted to the study of efficient numerical methods and software for solving the bidomain model; see, e.g. [6] for a review of bidomain solvers.

☆ The research reported was supported by the National Science and Engineering Research Council of Canada under grant number RGPIN 228090-2013.

* Corresponding author.
E-mail address: spiteri@cs.usask.ca (R.J. Spiteri).

http://dx.doi.org/10.1016/j.cam.2015.09.015
0377-0427/

In this paper, we revisit the behaviour of two popular first-order time-integration methods for solving the bidomain model. Both methods decouple the solution of the model at the cellular scale from that at the tissue scale in efforts to make the overall simulation more efficient. The first method is described in [7] and is loosely referred to here as the *semi-implicit method*; see also, e.g., [8–11] for similar usage. This method is implemented as the default bidomain solver in the Chaste software environment [9]. The second method is the first-order operator-splitting method described in [12]. The origin of this method dates back to [13] and is referred to here as the *Godunov method*. This method decouples the solution to the model at the cellular and tissue scales in a slightly different way than the semi-implicit method; details are provided in Section 2.2.

We present a comparative accuracy analysis for these two methods and find that, under conditions normally satisfied in practice, the error for the semi-implicit method is generically larger than that of the Godunov method. Provided numerical stability is not an issue, this suggests that the Godunov method can take larger time-steps than the semi-implicit method while meeting prescribed accuracy requirements. Because the cost per step of the Godunov method is comparable to that of the semi-implicit method, we hypothesize that the tradeoff in overall computation favours the Godunov method. This hypothesis is tested by comparing the computational performance of both methods to solve the bidomain model to typical accuracies required in practice. The methods are tested on four one-dimensional (1D) problems, a two-dimensional (2D) problem, and a three-dimensional (3D) problem. We find that indeed the Godunov method outperforms the semi-implicit method in every case considered, being between 5 and 15 times faster for the same level of accuracy.

The remainder of this paper is organized as follows. In Section 2, we briefly review the bidomain model, review the semi-implicit and Godunov methods as applied to the bidomain model, and perform a comparative error analysis on the methods. In Section 3, we present the numerical experiments and assess the performance of both methods for solving the bidomain model. Finally, in Section 4, we summarize our conclusions and give our recommendations for use of the methods.

2. Background

2.1. The bidomain model

The bidomain model is a multi-scale reaction–diffusion model that considers myocardial tissue as consisting of two co-located domains, the intracellular domain and the extracellular domain. The domains are separated by a non-conductive medium, so intracellular and extracellular currents cannot travel past the boundary of the heart. The bidomain model couples a system of reaction partial differential equations (PDEs) on the micro-scale (that eventually reduce to a system of non-spatially coupled ordinary differential equations (ODEs) at each point in the discretized domain) describing the chemical reactions and flow of ions across the cell membrane of individual myocardial cells with a system of diffusion PDEs on the macro-scale describing the propagation of the electrical activation through the heart. Because the bidomain model represents a continuum-based model, individual myocardial cells are generally not modelled. Rather, their net effect is captured in an averaged sense.

On a spatial domain $\Omega \subset \mathbb{R}^d$ of dimension d and time interval $[t_0, t_f]$, the bidomain model can be written as [12]

$$\frac{\partial \mathbf{s}}{\partial t} = \mathbf{f}(\mathbf{s}, v, t), \tag{1a}$$

$$\chi C_m \frac{\partial v}{\partial t} + \chi I_{\text{ion}}(\mathbf{s}, v, t) + I_{\text{stim}}(\mathbf{x}, t) = \nabla \cdot (\sigma_i \nabla v) + \nabla \cdot (\sigma_i \nabla u_e)\,, \tag{1b}$$

$$0 = \nabla \cdot (\sigma_i \nabla v) + \nabla \cdot ((\sigma_i + \sigma_e)\, \nabla u_e)\,, \tag{1c}$$

subject to boundary conditions on $\partial\Omega \times [t_0, t_f]$ given by

$$\begin{aligned} \hat{\mathbf{n}} \cdot (\sigma_i \nabla v + \sigma_i \nabla u_e) &= 0, \\ \hat{\mathbf{n}} \cdot (\sigma_e \nabla u_e) &= 0, \end{aligned} \tag{2}$$

where, at location $\mathbf{x} \in \Omega$ and time $t \in [t_0, t_f]$, $\mathbf{s} = \mathbf{s}(\mathbf{x}, t)$ is a vector related to the cellular states such as gating variables and ionic concentrations, $v = v(\mathbf{x}, t)$ is the transmembrane potential, and $u_e = u_e(\mathbf{x}, t)$ is the extracellular potential. The terms $\mathbf{f}(\mathbf{s}, \mathbf{v}, \mathbf{t})$ and $I_{\text{ion}}(\mathbf{s}, v, t)$ are non-linear functions related to the cell model; $I_{\text{stim}}(\mathbf{x}, t)$ is an applied stimulus current; σ_i and σ_e are conductivity tensors related to the intracellular and extracellular potentials, respectively. Finally, χ is the area of cell membrane per unit volume, C_m is the capacitance of the cell membrane per unit area, and $\hat{\mathbf{n}}$ is the unit outward normal to $\partial\Omega$. The boundary conditions represent the non-conductivity of the heart boundary.

2.2. Numerical methods and comparative error analysis

Although not always the optimal procedure [7], it is common to decouple the bidomain model into a system of non-spatially coupled ODEs at each spatial mesh point and a system of PDEs in order to compute the solution. Both the semi-implicit and Godunov methods considered in this study perform such a decoupling. The main difference between the methods is in their treatment of the I_{ion} term, which is nonlinear in v. The semi-implicit method includes the I_{ion} term

in the PDE system, rendering it formally nonlinear, whereas the Godunov method includes the I_{ion} term in the ODE systems, rendering the PDE system linear. We now describe both methods in detail and provide a comparative error analysis. In the analysis, both methods are assumed to employ the finite element method to spatially discretize the system of PDEs and a first-order explicit method to discretize the ODE systems. These choices are meant to reflect the implementation of the semi-implicit method as the default time-integration method within the Chaste software environment. For simplicity of exposition, we set $I_{\text{stim}} \equiv 0$ in the remainder of this section.

2.2.1. The semi-implicit method

In [8], Whiteley proposed a semi-implicit method based on the backward Euler method to discretize (1) in time but with the I_{ion} term evaluated at time t^n instead of t^{n+1}, thus avoiding nonlinearities in the discretization. Whiteley's method was modified in [7] to improve its efficiency by using a method similar to the forward Euler method replacing the backward Euler method used in [8] as the ODE solver. This method is implemented as the default time-integration method in Chaste. Specifically, the semi-implicit method used by Chaste as its default time-integration method to advance the numerical solution from time $t = t^n$ to time $t^{n+1} = t^n + \Delta t$ is a non-standard finite-difference method, see, e.g., [14], given by

$$\frac{\mathbf{s}^{n+1} - \mathbf{s}^n}{\Delta t} = \mathbf{f}(\mathbf{s}^n, v^n, t^n), \tag{3a}$$

$$\chi C_m \frac{v^{n+1} - v^n}{\Delta t} + \chi I_{\text{ion}}(\mathbf{s}^{n+1}, v^n, t^n) = \nabla \cdot \left(\sigma_i \nabla v^{n+1}\right) + \nabla \cdot \left(\sigma_i \nabla u_e^{n+1}\right), \tag{3b}$$

$$0 = \nabla \cdot \left(\sigma_i \nabla v^{n+1}\right) + \nabla \cdot \left((\sigma_i + \sigma_e)\, \nabla u_e^{n+1}\right), \tag{3c}$$

where Δt is the time step and superscripts indicate the time step index. Although not strictly necessary, Δt is generally constant in practice.

Standard application of the finite element method to discretize Eqs. (3b) and (3c) in space yields the fully discrete form

$$\mathbf{S}^{n+1} = \mathbf{S}^n + \Delta t\, \mathbf{f}(\mathbf{S}^n, \mathbf{v}^n, t^n), \tag{4a}$$

$$\begin{bmatrix} \frac{\chi C_m}{\Delta t}\mathbf{A} + \mathbf{A}_i & \mathbf{A}_i \\ \mathbf{A}_i & \mathbf{A}_{i+e} \end{bmatrix} \begin{bmatrix} \mathbf{v}^{n+1} \\ \mathbf{u}_e^{n+1} \end{bmatrix} = \begin{bmatrix} \frac{\chi C_m}{\Delta t}\mathbf{A}\mathbf{v}^n - \chi \mathbf{A}\mathbf{I}_{\text{ion}}(\mathbf{S}^{n+1}, \mathbf{v}^n, t^n) \\ 0 \end{bmatrix}, \tag{4b}$$

where $\mathbf{S}$, $\mathbf{v}$, and $\mathbf{u}_e$ are vectors representing the restrictions of v, u_e, and $\mathbf{s}$ to the discretized spatial domain and the elements of matrices $\mathbf{A}$, $\mathbf{A}_i$, and $\mathbf{A}_{i+e}$ are given by

$$\mathbf{A}(j,k) = \int_\Omega \phi_j\, \phi_k\, dx,$$

$$\mathbf{A}_i(j,k) = \int_\Omega \sigma_i\, \nabla\phi_j \cdot \nabla\phi_k\, dx,$$

$$\mathbf{A}_{i+e}(j,k) = \int_\Omega (\sigma_i + \sigma_e) \nabla\phi_j \cdot \nabla\phi_k\, dx,$$

where $j, k = 1, 2, \ldots, M$, M is the number of mesh points, and the ϕ are the finite element basis functions.

2.2.2. The Godunov method

The Godunov method is generally based on decomposing a system of ODEs into a number of sub-systems that are treated separately. Each sub-system is advanced in time, with advanced values of variables fed back into sub-systems that have not yet been advanced.

The implementation considered here for the bidomain model is described in [12]. The basic difference from the semi-implicit method (3) is that (1b) is split into two differential equations for v, a linear PDE of the form (1b) without the I_{ion} term, and a non-linear PDE involving only the I_{ion} term and no spatial derivatives. This second PDE (that reduces to an ODE at each mesh point) forms a sub-system with (1a) for $\mathbf{s}$.

Using a forward-Euler-like method to discretize the nonlinear system of ODEs and the backward Euler method to discretize the resulting linear system of PDEs in time yields the two sub-systems

$$\frac{\mathbf{s}^{n+1} - \mathbf{s}^n}{\Delta t} = \mathbf{f}(\mathbf{s}^n, v^n, t^n), \tag{5a}$$

$$\frac{\hat{v}^{n+1} - v^n}{\Delta t} = -\frac{1}{C_m} I_{\text{ion}}(\mathbf{s}^n, v^n, t^n), \tag{5b}$$

and

$$\chi C_m \frac{v^{n+1} - \hat{v}^{n+1}}{\Delta t} = \nabla \cdot \left(\sigma_i \nabla v^{n+1}\right) + \nabla \cdot \left(\sigma_i \nabla u_e^{n+1}\right), \tag{6a}$$

$$0 = \nabla \cdot \left(\sigma_i \nabla v^{n+1}\right) + \nabla \cdot \left(\left(\sigma_i + \sigma_e\right) \nabla u_e^{n+1}\right). \tag{6b}$$

The Godunov operator-splitting method is the implicit–explicit Runge–Kutta method [15] given by

$$\mathbf{S}^{n+1} = \mathbf{S}^n + \Delta t\, \mathbf{f}(\mathbf{S}^n, \mathbf{v}^n, t^n), \tag{7a}$$

$$\hat{\mathbf{v}}^{n+1} = \mathbf{v}^n - \frac{\Delta t}{C_m} \mathbf{I}_{\text{ion}}(\mathbf{S}^n, \mathbf{v}^n, t^n), \tag{7b}$$

followed by the update from solving

$$\begin{bmatrix} \frac{\chi C_m}{\Delta t}\mathbf{A} + \mathbf{A}_i & \mathbf{A}_i \\ \mathbf{A}_i & \mathbf{A}_{i+e} \end{bmatrix} \begin{bmatrix} \mathbf{v}^{n+1} \\ \mathbf{u}_e^{n+1} \end{bmatrix} = \begin{bmatrix} \frac{\chi C_m}{\Delta t}\mathbf{A}\hat{\mathbf{v}}^{n+1} \\ 0 \end{bmatrix}, \tag{8}$$

obtained by application of the finite element method to discretize (6), where $\mathbf{A}$, $\mathbf{A}_e$, and $\mathbf{A}_{i+e}$ are the matrices defined in (4).

Substituting (7b) into (8) yields

$$\mathbf{S}^{n+1} = \mathbf{S}^n + \Delta t\, \mathbf{f}(\mathbf{S}^n, \mathbf{v}^n, t^n), \tag{9a}$$

$$\begin{bmatrix} \frac{\chi C_m}{\Delta t}\mathbf{A} + \mathbf{A}_i & \mathbf{A}_i \\ \mathbf{A}_i & \mathbf{A}_{i+e} \end{bmatrix} \begin{bmatrix} \mathbf{v}^{n+1} \\ \mathbf{u}_e^{n+1} \end{bmatrix} = \begin{bmatrix} \frac{\chi C_m}{\Delta t}\mathbf{A}\mathbf{v}^n - \chi \mathbf{A}\mathbf{I}_{\text{ion}}(\mathbf{S}^n, \mathbf{v}^n, t^n) \\ 0 \end{bmatrix}. \tag{9b}$$

We note that (9a) is identical to (4a), and (4b) and (9b) only differ in the right-hand side vector. Assuming the costs for (iteratively) solving the linear systems (9b) and (4b) are comparable, the computational cost per step of the Godunov method is comparable to that of the semi-implicit method.

2.2.3. Comparative error analysis

We now present a comparative error analysis for the time-integration methods presented. For the purposes of the analysis, we assume that the conductivities are constant. In one dimension, we further define $\lambda = \sigma_e/\sigma_i$. Thus, we can use (1c) to write u_e in terms of v and simplify the bidomain model (1) to the monodomain model (see, e.g., [12]),

$$\frac{\partial \mathbf{s}}{\partial t} = \mathbf{f}(\mathbf{s}, v, t), \tag{10a}$$

$$\frac{\partial v}{\partial t} = \frac{\lambda}{1+\lambda} \frac{1}{\chi C_m} \nabla \cdot (\sigma_i \nabla v) - \frac{1}{C_m} I_{\text{ion}}(\mathbf{s}, v, t). \tag{10b}$$

If we apply the finite element method to (10b), then the spatially discrete system for the semi-implicit method takes the form

$$\mathbf{A}\frac{\partial \mathbf{v}_{h,SI}}{\partial t} = -\frac{\lambda}{1+\lambda} \frac{1}{\chi C_m} \mathbf{A}_i \mathbf{v}_{h,SI} - \frac{1}{C_m} \mathbf{A}\mathbf{I}_{\text{ion}}(\mathbf{S}, \mathbf{v}_{h,SI}, t), \tag{11}$$

and therefore, the fully discrete system for the semi-implicit method is

$$\frac{\mathbf{S}_{SI}^{n+1} - \mathbf{S}_{SI}^n}{\Delta t} = \mathbf{f}(\mathbf{S}_{SI}^n, \mathbf{v}_{SI}^n, t^n) \tag{12a}$$

$$\mathbf{A}\frac{\mathbf{v}_{SI}^{n+1} - \mathbf{v}_{SI}^n}{\Delta t} = -\frac{\lambda}{1+\lambda} \frac{1}{\chi C_m} \mathbf{A}_i \mathbf{v}_{SI}^{n+1} - \frac{1}{C_m} \mathbf{A}\mathbf{I}_{\text{ion}}(\mathbf{S}_{SI}^{n+1}, \mathbf{v}_{SI}^n, t^n). \tag{12b}$$

On the other hand, if we apply the Godunov splitting to (10b), then we have

$$\frac{\partial v}{\partial t} = -\frac{1}{C_m} I_{\text{ion}}(\mathbf{s}, v, t), \tag{13a}$$

$$\frac{\partial v}{\partial t} = \frac{\lambda}{1+\lambda} \frac{1}{\chi C_m} \nabla \cdot (\sigma_i \nabla v), \tag{13b}$$

and if we apply the finite element method to (13b), then the spatially discrete system for the Godunov method is

$$\mathbf{A}\frac{\partial \mathbf{v}_{h,G}}{\partial t} = -\frac{\lambda}{1+\lambda} \frac{1}{\chi C_m} \mathbf{A}_i \mathbf{v}_{h,G}, \tag{14}$$

and therefore, the fully discrete system for the Godunov method is

$$\frac{\mathbf{S}_G^{n+1}-\mathbf{S}_G^n}{\Delta t} = \mathbf{f}(\mathbf{S}_G^n, \mathbf{v}_G^n, t^n), \tag{15a}$$

$$\frac{\hat{\mathbf{v}}_G^{n+1}-\mathbf{v}_G^n}{\Delta t} = -\frac{1}{C_m}\mathbf{I}_{\text{ion}}(\mathbf{S}_G^n, \mathbf{v}_G^n, t^n), \tag{15b}$$

$$\mathbf{A}\frac{\mathbf{v}_G^{n+1}-\hat{\mathbf{v}}_G^{n+1}}{\Delta t} = -\frac{\lambda}{1+\lambda}\frac{1}{\chi C_m}\mathbf{A}_i\mathbf{v}_G^{n+1}. \tag{15c}$$

We begin with the temporal discretization error for (10a), which reduces to a system of ODEs at each mesh point. Both the semi-implicit and Godunov methods use the same (forward Euler) time discretization to advance (10a) for $\mathbf{s}$; hence (12a) and (15a) are identical and $\mathbf{v}$ is fixed to $\mathbf{v}^n$. Starting from the same initial condition $(\mathbf{S}^n, \mathbf{v}^n)$ for comparative purposes, the local truncation errors (see, e.g., [16]) of (12a) and (15a) are thus identical and equal to

$$\boldsymbol{\tau}_{\mathbf{s}} = \frac{\Delta t}{2}\left(\frac{\partial \mathbf{f}^n}{\partial t} + \frac{\partial \mathbf{f}^n}{\partial \mathbf{s}}\mathbf{f}^n\right) + O(\Delta t^2), \tag{16}$$

where $\mathbf{f}^n = \mathbf{f}(\mathbf{S}^n, \mathbf{v}^n, t^n)$.

Consequently, to compare the total error of the semi-implicit and Godunov methods, we only need to compare the error of $\mathbf{v}$ in (10b) for both methods. The error of the semi-implicit method includes contributions from the spatial and temporal discretizations of PDE (10b). The error of the Godunov method includes contributions from the splitting of PDE (10b), the spatial and temporal discretizations of PDE (13b), and the temporal discretization of ODE (13a).

If we suppose that $\mathbf{v}$ is the solution of (10b) restricted to the mesh defined by the spatial discretization, $\mathbf{v}_{h,SI}$ is the solution of (11), and $\mathbf{v}_{SI}$ is given by (12b), then the error of the semi-implicit method satisfies

$$\epsilon_{SI} = \|\mathbf{v}-\mathbf{v}_{SI}\|_{\ell^2} \le \left\|\mathbf{v}-\mathbf{v}_{h,SI}\right\|_{\ell^2} + \left\|\mathbf{v}_{h,SI}-\mathbf{v}_{SI}\right\|_{\ell^2}, \tag{17}$$

where $\|.\|_{\ell^2}$ is the 2-norm and $\left\|\mathbf{v}-\mathbf{v}_{h,SI}\right\|_{\ell^2}$ and $\left\|\mathbf{v}_{h,SI}-\mathbf{v}_{SI}\right\|_{\ell^2}$ can be interpreted as the spatial and temporal discretization errors of PDE (10b), respectively.

Similarly, if we suppose that $\mathbf{v}_{\text{split}}$ is the solution of (13b) restricted to the mesh, $\mathbf{v}_{h,G}$ is the solution of (14), and $\mathbf{v}^G$ is given by (15c), then the error of the Godunov method satisfies

$$\epsilon_G = \|\mathbf{v}-\mathbf{v}_G\|_{\ell^2} \le \left\|\mathbf{v}-\mathbf{v}_{\text{split}}\right\|_{\ell^2} + \left\|\mathbf{v}_{\text{split}}-\mathbf{v}_{h,G}\right\|_{\ell^2} + \left\|\mathbf{v}_{h,G}-\mathbf{v}_G\right\|_{\ell^2}, \tag{18}$$

where $\left\|\mathbf{v}-\mathbf{v}_{\text{split}}\right\|_{\ell^2}$ can be interpreted as the splitting error of PDE (10b) and $\left\|\mathbf{v}_{\text{split}}-\mathbf{v}_{h,G}\right\|_{\ell^2}$ as the spatial discretization error of PDE (13b) and $\left\|\mathbf{v}_{h,G}-\mathbf{v}_G\right\|_{\ell^2}$ includes the temporal discretization error of PDE (13b) and the temporal discretization error of ODE (13a).

We now compare the accuracy of the semi-implicit and Godunov methods by comparing ϵ_{SI} and ϵ_G. We consider a general analysis of the spatial discretization error introduced from the use of the finite element method; see, e.g., [17]. For simplicity of exposition, suppose that the coefficient of the diffusion term in (10b) is equal to unity.

Let S_h be a family of finite-dimensional subspaces of $L^2(\Omega)$ and T_h be a family of operators $T_h : L^2(\Omega) \to S_h$ approximating the exact solution operator $T : L^2(\Omega) \to H_0^1(\Omega)$ of the elliptic problem

$$\begin{aligned} \Delta u + g &= 0 \quad \text{in } \Omega, \\ \hat{\mathbf{n}}\cdot\nabla u &= 0 \quad \text{on } \partial\Omega; \end{aligned}$$

i.e., $u = Tg$, or $T = -(\Delta)^{-1}$, such that T_h is self-adjoint, positive semi-definite on $L^2(\Omega)$, and positive definite on S_h, and such that $\|(T_h - T)g\|_{L^2} \le Ch^s\|g\|_{L^2_{s-2}}$ with constant $C > 0$ independent of g and h, $2 \le s \le r$ for a positive integer $r \ge 2$, and $g \in H^{s-2}(\Omega)$, where

$$\|w\|_{L^2} = \left(\int_\Omega w^2 d\mathbf{x}\right)^{1/2}, \qquad \|w\|_{L^2_r} = \left(\sum_{|\alpha|\le r}\|D^\alpha w\|_{L^2}\right)^{1/2},$$

with $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_d)$ and $D^\alpha = (\frac{\partial}{\partial x_1})^{\alpha_1}(\frac{\partial}{\partial x_2})^{\alpha_2}\cdots(\frac{\partial}{\partial x_d})^{\alpha_d}$ denoting a derivative with respect to $\mathbf{x}$ of order $|\alpha| = \sum_{j=1}^d \alpha_j$; i.e., the sum consists of all such derivatives of order at most r. In the numerical experiments described below, the finite element method is of order of $O(h^2)$; i.e., $r = 2$.

If u_1 is now the solution of the time-dependent, non-homogeneous problem

$$\begin{aligned} \frac{\partial u_1}{\partial t} &= \Delta u_1 + g \quad \text{in } \Omega \times [0,\infty), \\ \hat{\mathbf{n}}\cdot\nabla u_1 &= 0 \quad \text{on } \partial\Omega \times [0,\infty), \\ u_1(\mathbf{x}, 0) &= u_1^0(\mathbf{x}) \quad \text{in } \bar{\Omega}, \end{aligned}$$

that corresponds with (10b) where $\bar{\Omega}$ is the closure of Ω and $u_{h,SI}$ is the solution of the semi-discrete problem

$$T_h \frac{\partial u_{h,SI}}{\partial t} = -u_{h,SI} + T_h g \quad \text{for } t \geq 0,$$
$$u_{h,SI}(0) = u^0_{h,SI},$$

that corresponds with Eq. (11), then following [17,18], it can be shown that

$$\|u_1(t) - u_{h,SI}(t)\|_{L^2} \leq \|u^0_1 - u^0_{h,SI}\|_{L^2} + 4Ch^r \|u^0_1\|_{L^2_r} + 6Ch^r \int_0^t \left\| \frac{\partial u_1(t')}{\partial t} \right\|_{L^2_r} dt'. \tag{19}$$

Similarly, if u_2 is the solution of the time-dependent, homogeneous equation

$$\frac{\partial u_2}{\partial t} = \Delta u_2 \quad \text{in } \Omega \times [0, \infty),$$
$$\hat{\mathbf{n}} \cdot \nabla u_2 = 0 \quad \text{on } \partial\Omega \times [0, \infty),$$
$$u_2(\mathbf{x}, 0) = u^0_2(\mathbf{x}) \quad \text{in } \bar{\Omega},$$

that corresponds with (13b) and $u_{h,G}$ is the solution of the semi-discrete problem

$$T_h \frac{\partial u_{h,G}}{\partial t} = -u_{h,G} \quad \text{for } t \geq 0,$$
$$u_{h,G}(0) = u^0_{h,G},$$

that corresponds with Eq. (14), then following [17,19], it can be shown that

$$\|u_2(t) - u_{h,G}(t)\|_{L^2} \leq 2\sqrt{3}Ch^r \left\| u^0_2 \right\|_{L^2_r}. \tag{20}$$

Assuming both methods start with the same initial condition and noting that $u^0_1 = v^0$ and $u^0_2 = \hat{v}^1$, then (5b) yields

$$\left\| u^0_2 \right\|_{L^2_r} \leq \left\| u^0_1 \right\|_{L^2_r} + \frac{\Delta t}{C_m} \left\| I^0_{\text{ion}} \right\|_{L^2_r}.$$

Therefore, (20) can be written as

$$\|u_2(t) - u_{h,G}(t)\|_{L^2} \leq 2\sqrt{3}Ch^r \left(\left\| u^0_1 \right\|_{L^2_r} + \frac{\Delta t}{C_m} \left\| I^0_{\text{ion}} \right\|_{L^2_r} \right), \tag{21}$$

where the constant C is the same as in (19).

Identifying $g = -I_{\text{ion}}/C_m$ allows us to write

$$\left\| \frac{\partial u_1(t')}{\partial t} \right\|_{L^2_r} \leq \|\Delta u_1(t')\|_{L^2_r} + \frac{1}{C_m} \|I_{\text{ion}}(t')\|_{L^2_r}.$$

Now assuming $\Delta t \leq t$, $\|\Delta u_1(t')\|_{L^2_r} \leq \|\Delta u_1\|^{\max}_{L^2_r}$, and $\|I_{\text{ion}}(t')\|_{L^2_r} \leq \|I_{\text{ion}}\|^{\max}_{L^2_r}$ for all $0 \leq t' \leq t$, we can write (19) as

$$\|u_1 - u_{h,SI}\| \leq \|u^0_1 - u^0_{h,SI}\|_{L^2} + 4Ch^r \|u^0_1\|_{L^2_r} + 6Ch^r \frac{t}{C_m} (\|\Delta u_1\|^{\max}_{L^2_r} + \|I_{\text{ion}}\|^{\max}_{L^2_r}), \tag{22}$$

and (21) as

$$\|u_2 - u_{h,G}\|_{L^2_r} \leq 2\sqrt{3}Ch^r \left(\left\| u^0_1 \right\|_{L^2_r} + \frac{t}{C_m} \|I_{\text{ion}}\|^{\max}_{L^2_r} \right). \tag{23}$$

We note that the error bounds (22) and (23) are given for continuous functions. However, analogous bounds are still valid for discrete functions using the 2-norm.

Because the bound (22) is generally larger than (23), we expect the spatial discretization error associated with the semi-implicit method to exceed that associated with the Godunov method. In practical calculations, the spatial error typically dominates the temporal error. This suggests that the semi-implicit method may generally be less accurate than the Godunov method in practice. A similar argument may be used to suggest that the temporal error of the semi-implicit method is also generally larger than that of the Godunov method.

To determine the temporal discretization error for the semi-implicit method, we find the local truncation error of (12b) to be

$$\boldsymbol{\tau}_{SI} = \frac{\Delta t}{2} \left[\left(\frac{\lambda}{1+\lambda} \frac{1}{\chi C_m} \right)^2 \mathbf{A}_i \mathbf{A}^{-1} \mathbf{A}_i \mathbf{v}^n + \frac{\lambda}{1+\lambda} \frac{1}{\chi C_m^2} \mathbf{A}_i \mathbf{I}^n_{\text{ion}} + \frac{1}{C_m} \mathbf{A} \frac{\partial \mathbf{I}^n_{\text{ion}}}{\partial t} - \frac{2}{C_m} \mathbf{A} \frac{\partial \mathbf{I}^n_{\text{ion}}}{\partial \mathbf{s}} \mathbf{f}^n \right] + O(\Delta t^2), \tag{24}$$

Table 1
Summary of the error analysis.

Error type	Semi-implicit method	Godunov method
Temporal discretization error for ODEs (10a)	Local truncation error (16)	Local truncation error (16)
Spatial discretization error for PDE ((10b) or (13b))	Error bound (22)	Error bound (23)
Splitting error for PDE (10b)	–	Local truncation error (25)
Temporal discretization error for PDE ((10b) or (13b))	Local truncation error (24)	Local truncation error (25)
Temporal discretization error for ODE (13a)	–	Local truncation error (25)

where for brevity $\mathbf{I}_{\text{ion}}^n := \mathbf{I}_{\text{ion}}(\mathbf{S}^n, \mathbf{v}^n, t^n)$. We note that the expression (24) captures the temporal discretization error of PDE (10b), i.e., $\left\|\mathbf{v}_{h,SI} - \mathbf{v}_{SI}\right\|_{\ell^2}$.

Similarly, for the Godunov method, using (15b), the local truncation error of (15c) is

$$\boldsymbol{\tau}_G = \frac{\Delta t}{2}\left[\left(\frac{\lambda}{1+\lambda}\frac{1}{\chi C_m}\right)^2 \mathbf{A}_i\mathbf{A}^{-1}\mathbf{A}_i\mathbf{v}^n + \frac{\lambda}{1+\lambda}\frac{1}{\chi C_m^2}\mathbf{A}_i\mathbf{I}_{\text{ion}}^n + \frac{1}{C_m}\mathbf{A}\frac{\partial \mathbf{I}_{\text{ion}}^n}{\partial t}\right] + O(\Delta t^2). \tag{25}$$

Application of the triangle inequality immediately yields that the bound on $\|\boldsymbol{\tau}_{SI}\|$ is larger than that for $\|\boldsymbol{\tau}_G\|$. Thus, it may be reasonable to expect $\|\boldsymbol{\tau}_{SI}\|$ to exceed $\|\boldsymbol{\tau}_G\|$, and the numerical experiments presented below show that this is true in practical cases using $\|\cdot\|_{\ell^2}$.

A summary of the contributions to the errors in the semi-implicit and Godunov methods is presented in Table 1. Our analysis has suggested that the spatial and temporal discretizations of the semi-implicit method can be expected to be larger than those of the Godunov method. Consequently, we expect the semi-implicit method to be less accurate than the Godunov method for a given discretization. Put differently, for a given spatial discretization, the semi-implicit method requires a smaller time step than the Godunov method to achieve a given level of accuracy.

To determine which method may ultimately be more efficient in practice, the computational cost per step of each method must be considered. As we mentioned earlier, the semi-implicit and Godunov methods have comparable computational costs per step. Consequently, the increase in Δt afforded to the Godunov method through increased accuracy renders it more efficient than the semi-implicit method. This is demonstrated in the next section using a diverse set of numerical experiments.

3. Numerical experiments

In this section, we assess the performance of the semi-implicit and Godunov methods by means of numerical experiments consisting of four one-dimensional problems, a two-dimensional problem, and a three-dimensional problem. These experiments include two different combinations of initial conditions and stimulus and four different cell models with a wide range of stiffness characteristics. The Chaste software environment [9] is used for all of the numerical tests. The (Chaste default) interpolation approach, i.e., ionic current interpolation, is used to interpolate the I_{ion} term as required in the finite element method [10]. To solve the linear systems associated with each method, the default pair of Krylov subspace (KSP) solver and preconditioner in Chaste are used, i.e., the conjugate gradient solver with block Jacobi preconditioner. Timings were computed in Chaste 3.1 running in serial on a dual Hex Core Intel Xeon X5650 2.66 GHz with 12 GB of RAM running Red Hat Enterprise Linux Server 5.7. Timings (both absolute and relative) are naturally affected by any parallelism introduced in the computations, e.g., the number of cores, available memory, and specific architecture, especially as these relate to the relative expense of solving (1a) versus (1b)–(1c). However, a comprehensive study of these issues is beyond the scope of this paper.

In order to assess the accuracy of the semi-implicit and Godunov methods, reference solutions are generated in Chaste using the semi-implicit method but with Heun's method (instead of forward Euler) for advancing the **s** variables. By comparing solutions computed by halving the time step and doubling the number of mesh points, reference solutions with two to four matching digits between successive solutions were computed for each numerical experiment. Solutions were compared at $N_t = 21$ equally spaced points in the time interval and $N_x = 101$ equally spaced points in the space interval, for a total of $N = N_t N_x = 2121$ space–time points.

To calculate the accuracy of the numerical solutions, we use the Mixed Root Mean Square (MRMS) error [20], defined by

$$[\text{MRMS}]_w = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{\bar{w}_i - w_i}{1+|\bar{w}_i|}\right)^2},$$

where $\bar{w}_i$ and w_i respectively denote the reference solution and the numerical solution for the (scalar) quantity w at space–time point i.

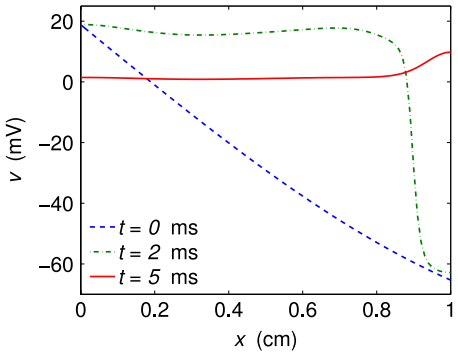


Fig. 1. Transmembrane potential from the Godunov method at $t = 0$ ms (– – –), $t = 2$ ms (·–·–·), and $t = 5$ ms (——) for the bidomain model coupled with the cell model of Courtemanche et al. with (26).

3.1. 1D experiments

The bidomain model was simulated in one dimension with two cell models, namely, those of Courtemanche et al. [21] and Winslow et al. [22]. The model of Courtemanche et al. has 21 cellular state variables and is considered to be moderately stiff, whereas the model of Winslow et al. has 31 variables and is considered to be highly stiff [20,23].

Two different combinations of initial conditions and stimulus were used. The first is given by

$$\begin{aligned} v(x,0) &= v_0 + 100(1-\sin(x)),\\ \mathbf{s}(x,0) &= \mathbf{s}_0,\\ u_e(x,0) &= 0,\\ I_{\text{stim}}(x,t) &= 0, \quad x \in [0,1]\,\text{cm},\ t \in [0,t_f]\,\text{ms}, \end{aligned} \tag{26}$$

where v_0 and $\mathbf{s}_0$ are the default resting state values for v and $\mathbf{s}$, respectively, for the particular cell model used [24]. The second is given by

$$\begin{aligned} v(x,0) &= v_0,\\ \mathbf{s}(x,0) &= \mathbf{s}_0,\\ u_e(x,0) &= 0,\\ I_{\text{stim}}(x,t) &= \begin{cases} -50\,000\ \mu\text{A/cm}^3, & x \in [0,0.1]\,\text{cm},\ t \in [0,2]\,\text{ms},\\ 0, & \text{otherwise.} \end{cases} \end{aligned} \tag{27}$$

For all the simulations, the (Chaste default) parameter values $\chi = 1400/\text{cm}$, $C_m = 1\ \mu\text{F/cm}^2$, $\sigma_i = 1.75$ mS/cm, and $\sigma_e = 7$ mS/cm were used, and the spatial domain was [0, 1] cm. The time interval was [0, 5] ms for the model of Courtemanche et al. and is [0, 10] ms for the model of Winslow et al.

Two reference solutions with four matching digits between successive numerical solutions were generated using $\Delta t = 5 \cdot 10^{-7}$ ms with 20,001 mesh points for the model of Courtemanche et al. and 50,001 mesh points for the model of Winslow et al. The maximum constant time step and minimum number of mesh points that yielded a numerical solution that approximately satisfied a 5% MRMS error tolerance for v were found with both the semi-implicit and Godunov methods; the MRMS errors for u_e were found to be smaller (see Table 2).

The plots of the transmembrane potential obtained from the Godunov method for the model of Courtemanche et al. at $t = 0$ ms, $t = 2$ ms, and $t = 5$ ms using (26) and (27) are given in Figs. 1 and 2, respectively. The plots of the transmembrane potential obtained from the Godunov method for the model of Winslow et al. at $t = 0$ ms, $t = 5$ ms, and $t = 10$ ms using (26) and (27) are given in Figs. 3 and 4, respectively. The corresponding plots for the reference solution and the solution from the semi-implicit method are indistinguishable to normal visual accuracy and are thus omitted.

For the combinations of initial conditions and stimulus (26) and (27), Figs. 5 and 6 show normalized local truncation errors (24) and (25) for the model of Courtemanche et al., respectively, and Figs. 7 and 8 show normalized local truncation errors (24) and (25) for the model of Winslow et al., respectively. These figures were produced using the typical practical values $\Delta t = 1 \cdot 10^{-2}$ ms and $N_x^* = 201$ values of the reference solution where required and normalized by N_x^*. From these figures, we see that the size of the normalized local truncation error of the semi-implicit method is never less than that of the Godunov method. Combined with the analysis from Section 2.2, this suggests the Godunov method may outperform the semi-implicit method.

The maximum time steps used to obtain the solutions with approximately 5% MRMS error are reported in Table 2. For the model of Courtemanche et al., we used 101 mesh points for the case with (26) and 401 mesh points with (27). For the model of Winslow et al., we used 251 mesh points for the case with (26) and 1001 mesh points with (27).

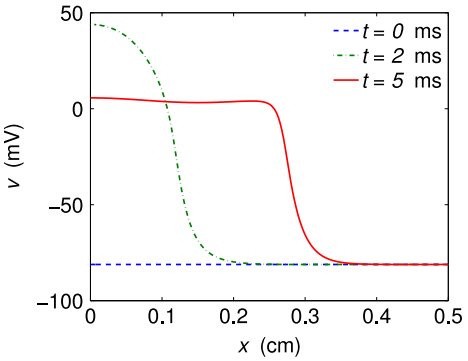


Fig. 2. Transmembrane potential from the Godunov method at $t = 0$ ms (– – –), $t = 2$ ms (·–·–·), and $t = 5$ ms (——) for the bidomain model coupled with the cell model of Courtemanche et al. with (27).

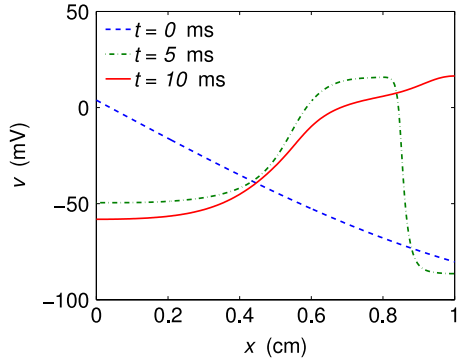


Fig. 3. Transmembrane potential of the Godunov solution at $t = 0$ ms (– – –), $t = 5$ ms (·–·–·), and $t = 10$ ms (——) for the bidomain model coupled with the cell model of Winslow et al. with (26).

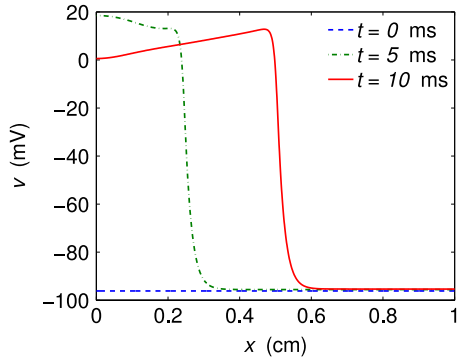


Fig. 4. Transmembrane potential of the Godunov solution at $t = 0$ ms (– – –), $t = 5$ ms (·–·–·), and $t = 10$ ms (——) for the bidomain model coupled with the cell model of Winslow et al. with (27).

Table 2 also shows the error and timing results, from which we see that the Godunov (G) method can take a larger time step than the semi-implicit (SI) method[1] and provide less than 5% MRMS error for v. The Godunov method is 10.2 and 10 times faster than the semi-implicit method for the model of Courtemanche et al. with (26) and (27), respectively, and 9.8 and 9.7 times faster for the model of Winslow et al. with (26) and (27), respectively. We note that approximately 80% to 90% of the total execution time is spent on advancing the cell states via (3a) for these problems.

Similar experiments to satisfy a 5% MRMS error tolerance on v were performed in 1D with (26) for other cell models including the model of Maleckar et al. [25], the Luo–Rudy phase 1 (LR1) model [26], the model of Noble et al. 1998 [27], and the epicardial variant of the model of ten Tusscher et al. (2006) (TT2006-epi) [28]. The results showed that the Godunov method outperforms the semi-implicit method in all of these cases by factors of 15, 6, 8, and 9, respectively; full details are omitted.

[1] Specifically, ten times larger for the problems tested.

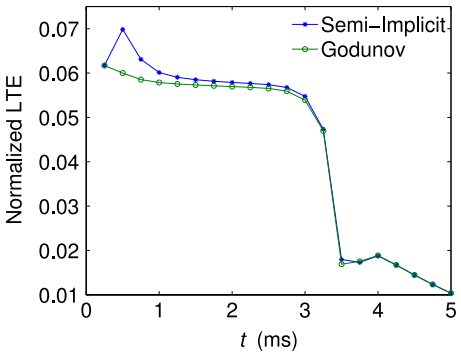


Fig. 5. Normalized local truncation errors (24) and (25) for the bidomain model coupled with the cell model of Courtemanche et al. with (26).

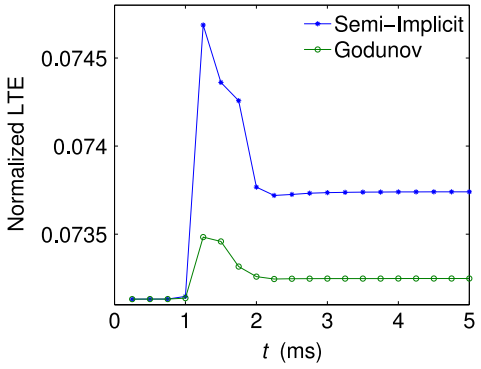


Fig. 6. Normalized local truncation errors (24) and (25) for the bidomain model coupled with the cell model of Courtemanche et al. with (27).

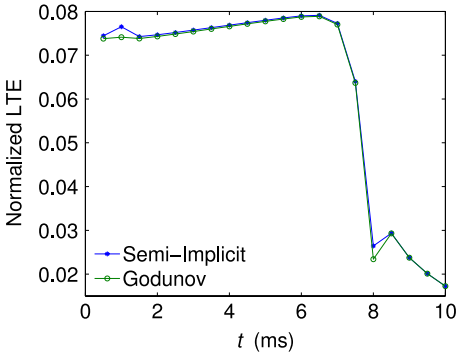


Fig. 7. Normalized local truncation errors (24) and (25) for the bidomain model coupled with the cell model of Winslow et al. with (26).

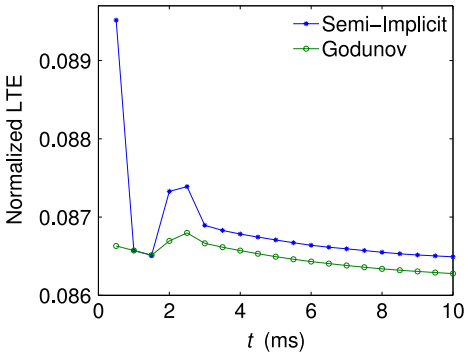


Fig. 8. Normalized local truncation errors (24) and (25) for the bidomain model coupled with the cell model of Winslow et al. with (27).

Table 2
Time step, in milliseconds, MRMS errors, and execution time, in seconds, of the semi-implicit and Godunov methods for the bidomain model coupled with the cell models of Courtemanche et al. and the model of Winslow et al. over the spatial domain [0, 1].

Cell model	Method	Δt (ms)	$[\text{MRMS}]_v$	$[\text{MRMS}]_{u_e}$	Execution time (s)
Courtemanche et al. with (26)	SI	$1 \cdot 10^{-3}$	0.048	0.022	2.45
	G	$1 \cdot 10^{-2}$	0.033	0.028	0.24
Courtemanche et al. with (27)	SI	$1 \cdot 10^{-4}$	0.042	0.020	52.99
	G	$1 \cdot 10^{-3}$	0.040	0.018	5.30
Winslow et al. with (26)	SI	$1 \cdot 10^{-5}$	0.031	0.020	477.99
	G	$1 \cdot 10^{-4}$	0.029	0.019	50.77
Winslow et al. with (27)	SI	$1 \cdot 10^{-5}$	0.037	0.011	1846.58
	G	$1 \cdot 10^{-4}$	0.034	0.010	189.41

Table 3
Time step, in milliseconds, MRMS errors, and execution time, in seconds, of the semi-implicit and Godunov methods for the bidomain model coupled with the LR1 cell model over the spatial domain [0, 1] cm × [0, 1] cm.

Method	Δt (ms)	$[\text{MRMS}]_v$	$[\text{MRMS}]_{u_e}$	Execution time (s)
SI	$2 \cdot 10^{-3}$	0.033	0.009	38.87
G	$1 \cdot 10^{-2}$	0.030	0.033	8.47

3.2. 2D experiment

The bidomain model coupled with the LR1 cell model [26] was simulated to assess the performance of the semi-implicit and Godunov methods over a two-dimensional domain with the initial condition,

$$\begin{aligned} v(x, y, 0) &= v_0 + 100(1 - \sin(xy)), \\ \mathbf{s}(x, y, 0) &= \mathbf{s}_0, \\ u_e(x, y, 0) &= 0, \end{aligned} \tag{28}$$

where v_0 and $\mathbf{s}_0$ are the default resting state values for v and $\mathbf{s}$, respectively, for the LR1 cell model. The LRI cell model is considered to be relatively non-stiff [20,23]. The (Chaste default) parameter values $\chi = 1400/\text{cm}$, $C_m = 1\ \mu\text{F/cm}^2$, and $\sigma_i = \text{diag}(\sigma_i^f, \sigma_i^n)$, $\sigma_e = \text{diag}(\sigma_e^f, \sigma_e^n)$, with $\sigma_i^f = \sigma_i^n = 1.75$ mS/cm, $\sigma_e^f = \sigma_e^n = 7$ mS/cm were used, the spatial domain was [0, 1] cm × [0, 1] cm with 101 × 101 mesh points, and the time interval was [0, 5] ms.

A reference solution with two matching digits was generated using $\Delta t = 5 \cdot 10^{-6}$ ms and 1001 × 1001 mesh points. The maximum constant time step and minimum number of mesh points that yielded a numerical solution that approximately satisfied a 5% MRMS error tolerance for v were found with both the semi-implicit and Godunov methods; the MRMS errors for u_e were found to be smaller (see Table 3). The plots of the transmembrane potential of the Godunov solution at $t = 0$ ms, $t = 1$ ms, and $t = 5$ ms are given in Fig. 9. The corresponding plots for the reference solution and the semi-implicit method are indistinguishable to normal visual accuracy and are thus omitted.

Normalized values of the local truncation errors (24) and (25) were computed for the analogous 1D problem with (26). The plots of the normalized local truncation error over time show that the size of the normalized local truncation error of the semi-implicit method is never less than that of the Godunov method; full details are omitted. Combined with the analysis from Section 2.2, this suggests the Godunov method may outperform the semi-implicit method.

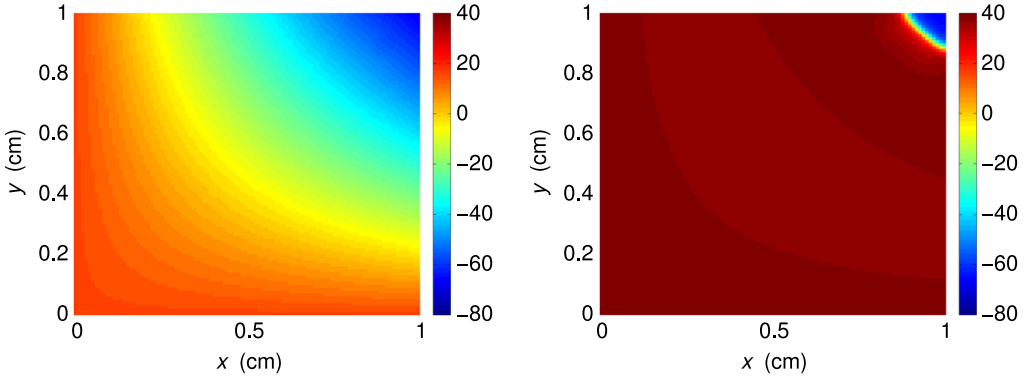
The maximum time steps used to obtain the solutions with approximately 5% MRMS error are reported in Table 3. Table 3 also shows the error and timing results, from which we see that the Godunov method can again use a larger time step (five times larger) than the semi-implicit method to provide less than 5% MRMS error for v. Consequently, the Godunov method is 4.6 times faster than the semi-implicit method for the LR1 cell model over the two-dimensional domain. We note that approximately 40%–45% of the total execution time is spent on advancing the cell states via (3a) for this problem.

A similar experiment to satisfy a 5% MRMS error tolerance on v was performed for this problem with (27) with the stimulus region extended to 2D, i.e., [0, 0.1] cm × [0, 0.1] cm. This experiment, which is similar to the experiment in [29], also showed that Godunov method is more efficient than the semi-implicit method; full details are omitted.

3.3. 3D experiment

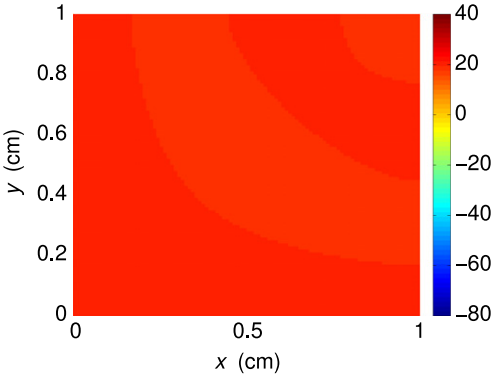
The bidomain model combined with TT2006-epi was simulated to assess the performance of the semi-implicit method and the Godunov method over a three-dimensional domain with the initial condition,

$$\begin{aligned} v(x, y, z, 0) &= v_0 + 100(1 - \sin(xyz)), \\ \mathbf{s}(x, y, z, 0) &= \mathbf{s}_0, \\ u_e(x, y, z, 0) &= 0, \end{aligned} \tag{29}$$



(a) Transmembrane potential at $t = 0$ ms.

(b) Transmembrane potential at $t = 1$ ms.



(c) Transmembrane potential at $t = 5$ ms.

Fig. 9. Transmembrane potential of the Godunov solution for the bidomain model coupled with the LR1 cell model at $t = 0$ ms, $t = 1$ ms, and $t = 5$ ms on the spatial domain $[0, 1]$ cm $\times$ $[0, 1]$ cm.

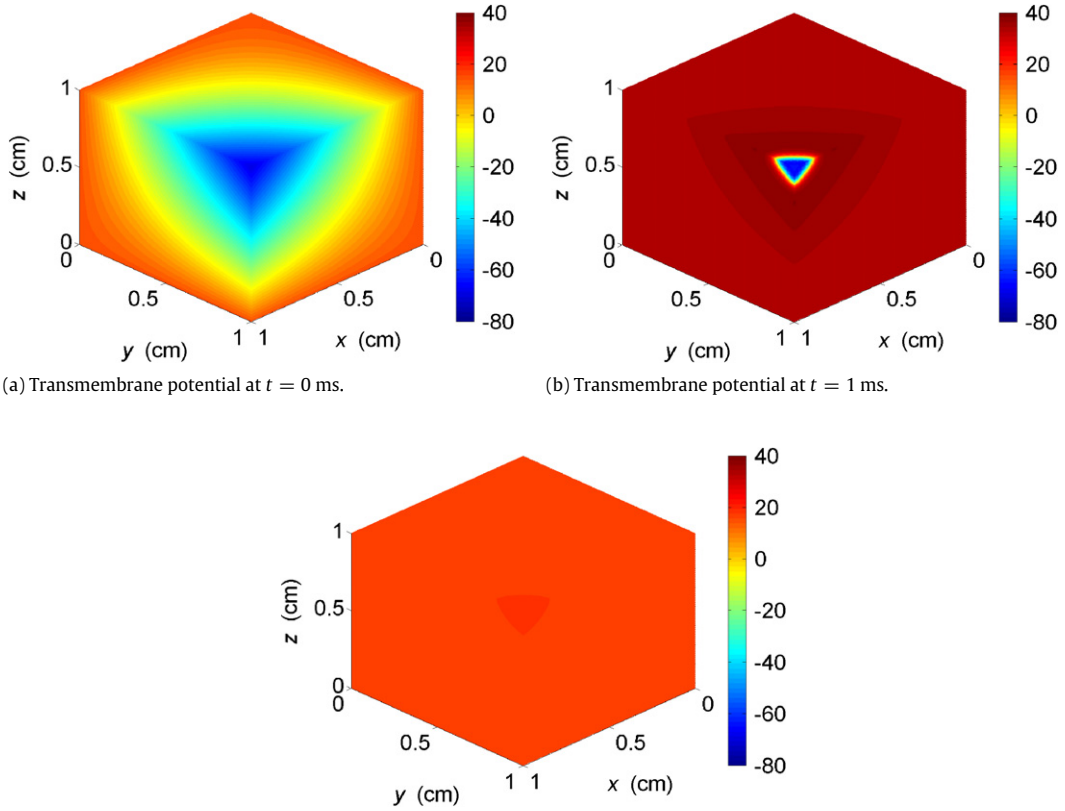
where v_0 and $\mathbf{s}_0$ are the default resting state values for v and $\mathbf{s}$, respectively, for TT2006-epi. TT2006-epi is considered to be a highly stiff cell model [20,23]. The (Chaste default) parameter values $\chi = 1400/\text{cm}$, $C_m = 1\,\mu\text{F/cm}^2$, $\sigma_i = \text{diag}(\sigma_i^f, \sigma_i^n, \sigma_i^t)$, $\sigma_e = \text{diag}(\sigma_e^f, \sigma_e^n, \sigma_e^t)$, with $\sigma_i^f = \sigma_i^n = \sigma_i^t = 1.75$ mS/cm, $\sigma_e^f = \sigma_e^n = \sigma_e^t = 7$ mS/cm were used, the spatial domain was $[0, 1]$ cm $\times$ $[0, 1]$ cm $\times$ $[0, 1]$ cm with $51 \times 51 \times 51$ mesh points, and the time interval was $[0, 5]$ ms.

Because of the size of the problem, we were unable to generate a reference solution using Chaste. Instead, a reference solution was generated with two matching digits in Nektar++ [30], which is a code based on spectral elements, using the following protocol. First, seven solutions were generated with polynomial degree 11 and time steps $\Delta t = 2^{-k} \cdot 10^{-2}$ ms, $k = -1, 0, \ldots, 5$. The Strang operator splitting method [31] together with the spectral/*hp* element method [30] for the spatial discretization of the PDE system (1b)–(1c), the first-order IMEX Gear method [32] for the temporal discretization of (1b)–(1c), and the forward Euler method for (1a) were used in Nektar++ to generate the seven solutions. Richardson extrapolation [33] was then used to generate the reference solution.

The maximum constant time step and minimum number of mesh points that yielded a numerical solution that approximately satisfied a 5% MRMS error tolerance for v were found with both the semi-implicit and Godunov methods; the MRMS errors for u_e were found to be smaller (see Table 4). The plots of the transmembrane potential of the Godunov solution at $t = 0$ ms, $t = 1$ ms, and $t = 5$ ms are given in Fig. 10. The corresponding plots for the reference solution and the semi-implicit are visually indistinguishable at normal resolutions and hence are omitted.

Normalized values of the local truncation errors (24) and (25) were computed for the analogous 1D problem with (26). The resulting plots of normalized local truncation error over time show that the size of the normalized local truncation error of the semi-implicit method is never less than that of the Godunov method; full details are omitted. Combined with the analysis from Section 2.2, this suggests the Godunov method may outperform the semi-implicit method.

The maximum time steps used to obtain the solutions with approximately 5% MRMS error are reported in Table 4. Table 4 also shows the error and timing results, from which we see that the Godunov method can again use a larger time step (ten times larger) than the semi-implicit method to provide less than 5% MRMS error for v. Consequently, the Godunov method is 7.7 times faster than the semi-implicit method for TT2006-epi over the three-dimensional domain. We note that approximately 40%–55% of the total execution time is spent on advancing the cell states (3a) for this problem.



(a) Transmembrane potential at $t = 0$ ms.

(b) Transmembrane potential at $t = 1$ ms.

(c) Transmembrane potential at $t = 5$ ms.

Fig. 10. Transmembrane potential of the Godunov solution for the bidomain model coupled with the TT2006-epi cell model at $t = 0$ ms, $t = 1$ ms, and $t = 5$ ms on the spatial domain $[0, 1] \times [0, 1] \times [0, 1]$ cm.

Table 4
Time step, in milliseconds, MRMS errors, and execution time, in seconds, of the semi-implicit and Godunov methods for the bidomain model coupled with the TT2006-epi cell model over the spatial domain $[0, 1]$ cm $\times$ $[0, 1]$ cm $\times$ $[0, 1]$ cm.

Method	Δt (ms)	$[\text{MRMS}]_v$	$[\text{MRMS}]_{u_e}$	Execution time (s)
SI	$2.5 \cdot 10^{-3}$	0.028	0.004	1232.88
G	$2.5 \cdot 10^{-2}$	0.035	0.044	160.32

A similar experiment to satisfy a 5% MRMS error tolerance on v was performed for this problem with (27) with the stimulus region extended to 3D, i.e., $[0, 0.1]$ cm $\times$ $[0, 0.1]$ cm $\times$ $[0, 0.1]$ cm. This experiment, which is similar to the experiment in [34], also showed that Godunov method is more efficient than the semi-implicit method; full details are omitted.

4. Conclusions

The numerical solution of the bidomain model is computationally demanding. Accordingly, the efficiency of the numerical methods is of paramount importance. In this paper, we have presented analysis that suggests that the semi-implicit method, which is part of the default bidomain solver in the prominent Chaste software environment, is generally not as efficient as the Godunov operator-splitting method under typical conditions in practice. This hypothesis is demonstrated to hold on a comprehensive set of test problems in one, two, and three dimensions, where it is observed that the Godunov method is between 5 and 15 times faster than the semi-implicit method for the same level of accuracy in the cases considered.

Consequently, we generally recommend the use of the Godunov method over the semi-implicit method for the numerical solution of the bidomain for typical practical accuracies of around 5% MRMS error or in situations where numerical stability is not a limiting factor.

A rigorous investigation as to the conditions under which the local truncation error (24) of the semi-implicit method is larger than (25) of the Godunov method is left as future work.

Acknowledgement

The authors wish to express their thanks to Megan E. Marsh for her help with Chaste.

References

[1] World Health Organization, World Health Organization: The 10 leading causes of death in the world, 2000 and 2012, May 2014. http://www.who.int/mediacentre/factsheets/fs310/en/.
[2] S.L. Murphy, J.Q. Xu, K.D. Kochanek, Deaths: Final data for 2010, Natl. Vital Stat. Rep. 61 (4) (2013) 1–118.
[3] P.A. Heidenreich, J.G. Trogdon, O.A. Khavjou, et al., Forecasting the future of cardiovascular disease in the United States: a policy statement from the American Heart Association, Circulation 123 (8) (2011) 933–944.
[4] Heart and Stroke Foundation of Canada, October 2002. [link]. URL: http://www.heartandstroke.com/site/c.ikIQLcMWJtE/b.5052135/k.2C86/Heart_disease__Atrial_fibrillation.htm.
[5] L. Tung, A bi-domain model for describing ischemic myocardial d-c potentials (Ph.D. thesis), Massachusetts Institute of Technology, 1978.
[6] E.J. Vigmond, R.W. dos Santos, A.J. Prassl, M. Deo, G. Plank, Solvers for the cardiac bidomain equations, Prog. Biophys. Mol. Biol. 96 (1–3) (2008) 3–18.
[7] J.A. Southern, G. Plank, E.J. Vigmond, J.P. Whiteley, Solving the coupled system improves computational efficiency of the bidomain equations, IEEE Trans. Biomed. Eng. 56 (10) (2009) 2404–2412.
[8] J.P. Whiteley, An efficient numerical technique for the solution of the monodomain and bidomain equations, IEEE Trans. Biomed. Eng. 53 (2006) 2139–2147.
[9] J. Pitt-Francis, P. Pathmanathan, M. Bernabeu, R. Bordas, J. Cooper, A. Fletcher, G. Mirams, P. Murray, J. Osborne, A. Walter, S. Chapman, A. Garny, I. van Leeuwen, P. Maini, B. Rodriguez, S. Waters, J. Whiteley, H. Byrne, D. Gavaghan, Chaste: A test-driven approach to software development for biological modelling, Comput. Phys. Comm. 180 (12) (2009) 2452–2471.
[10] P. Pathmanathan, G.R. Mirams, J. Southern, J.P. Whiteley, The significant effect of the choice of ionic current integration method in cardiac electrophysiological simulations, Int. J. Numer. Methods Biomed. Eng. 27 (2011) 1751–1770.
[11] P. Pathmanathan, M.O. Bernabeu, S.A. Niederer, D.J. Gavaghan, D. Kay, Computational modelling of cardiac electrophysiology: explanation of the variability of results from different numerical solvers, Int. J. Numer. Methods Biomed. Eng. 28 (2012) 890–903.
[12] G.T. Sundnes, J. Lines, X. Cai, B.F. Nielsen, K.A. Mardal, A. Tveito, Computing the Electrical Activity in the Heart, Springer-Verlag, Berlin, 2006.
[13] S.K. Godunov, A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics, Math. Sb. 47 (89) (1959) 271–306.
[14] R.E. Mickens, Nonstandard Finite Difference Models of Differential Equations, World Scientific Publishing Co., Inc., River Edge, NJ, 1994.
[15] U.M. Ascher, S.J. Ruuth, R.J. Spiteri, Implicit–explicit Runge–Kutta methods for time-dependent partial differential equations, Appl. Numer. Math. 25 (2–3) (1997) 151–167.
[16] A. Iserles, A First Course in the Numerical Analysis of Differential Equations, second ed., in: Cambridge Texts in Applied Mathematics, Cambridge University Press, Cambridge, 2008.
[17] V. Thomee, Galerkin Finite Element Methods for Parabolic Problems, second ed., in: Springer Series in Computational Mathematics, vol. 25, Springer, Berlin, 2006.
[18] V. Thomee, Negative norm estimates and superconvergence in Galerkin methods for parabolic problems, Math. Comp. 34 (149) (1980) 93–113.
[19] J.H. Bramble, A.H. Schatz, V. Thomee, L.B. Wahlbin, Some convergence estimates for semidiscrete Galerkin type approximations for parabolic equations, SIAM J. Numer. Anal. 14 (1977) 299–349.
[20] M.E. Marsh, S. Torabi Ziaratgahi, R.J. Spiteri, The secrets to the success of the Rush–Larsen method and its generalizations, IEEE Trans. Biomed. Eng. 59 (9) (2012) 2506–2515.
[21] M. Courtemanche, R. Ramirez, S. Nattel, Ionic mechanisms underlying human atrial action potential properties: insights from a mathematical model, Amer. J. Physiol. 275 (1) (1998) H301–H321.
[22] R.L. Winslow, J. Rice, S. Jafri, E. Marbán, B. O'Rourke, Mechanisms of altered excitation-contraction coupling in canine tachycardia-induced heart failure, II: Model studies, Circ. Res. 84 (5) (1999) 571–586.
[23] R.J. Spiteri, R.C. Dean, Stiffness analysis of cardiac electrophysiological models, Ann. Biomed. Eng. 38 (12) (2010) 3592–3604.
[24] A.B. Institute, The CellML project, November 2011. http://www.cellml.org/.
[25] M.M. Maleckar, J.L. Greenstein, N.A. Trayanova, W.R. Giles, Mathematical simulations of ligand-gated and cell-type specific effects on the action potential of human atrium, Prog. Biophys. Mol. Biol. 98 (2–3) (2008) 161–170.
[26] C. Luo, Y. Rudy, A model of ventricular cardiac action potential, Circ. Res. 68 (6) (1991) 1501–1526.
[27] D. Noble, A. Varghese, P. Kohl, P. Noble, Improved guinea-pig ventricular cell model incorporating a diadic space, IKr and IKs, and length- and tension-dependent processes, Canad. J. Cardiol. 14 (1) (1998) 123–134.
[28] K.H.W.J. ten Tusscher, A.V. Panfilov, Alternans and spiral breakup in a human ventricular tissue model, Amer. J. Physiol. Heart Circ. Physiol. 291 (3) (2006) H1088–H1100.
[29] S. Torabi Ziaratgahi, M.E. Marsh, J. Sundnes, R.J. Spiteri, Stable time integration suppresses unphysical oscillations in the bidomain model, Front. Phys. 2 (2014) 40.
[30] Nektar, Reference manual to Nektar++ library, 2012. http://www.nektar.info/wiki/3.2/Reference.
[31] G. Strang, On the construction and comparison of difference schemes, SIAM J. Numer. Anal. 5 (1) (1968) 506–517.
[32] M. Ethier, Y. Bourgault, Semi-implicit time-discretization schemes for the bidomain model, SIAM J. Numer. Anal. 46 (5) (2008) 2443–2468. http://dx.doi.org/10.1137/070680503.
[33] L.F. Richardson, The approximate arithmetical solution by finite differences of physical problems including differential equations, with an application to the stresses in a masonry dam, Phil. Trans. R. Soc. A 210 (1911) 307–357.
[34] S.A. Niederer, E. Kerfoot, A.P. Benson, M.O. Bernabeu, C.B.O. Bernus, E.M. Cherry, F.H.F.R. Clayton, E.H.A. Garny, S. Land, M. Maleckar, P. Pathmanathan, G. Plank, J.F. Rodriguez, I. Roy, F.B. Sachse, G. Seemann, O. Skavhaug, N.P. Smith, Verification of cardiac tissue electrophysiology simulators using an *n*-version benchmark, Phil. Trans. R. Soc. A 369 (2011) 4331–4351.