The Secrets to the Success of the Rush–Larsen Method and its Generalizations

Megan E. Marsh, Saeed Torabi Ziaratgahi, and Raymond J. Spiteri*

***Abstract*—One of the most popular methods for solving the ordinary differential equations (ODEs) that describe the dynamic behavior of myocardial cell models is known as the Rush–Larsen (RL) method. Its popularity stems from its improved stability over integrators such as the forward Euler (FE) method along with its easy implementation. The RL method partitions the ODEs into two sets: one for the gating variables, which are treated by an exponential integrator, and another for the remaining equations, which are treated by the FE method. The success of the RL method can be understood in terms of its relatively good stability when treating the gating variables. However, this feature would not be expected to be of benefit on cell models for which the stiffness is not captured by the gating equations. We demonstrate that this is indeed the case on a number of stiff cell models. We further propose a new partitioned method based on the combination of a first-order generalization of the RL method with the FE method. This new method leads to simulations of stiff cell models that are often one or two orders of magnitude faster than the original RL method.**

***Index Terms*—Efficient numerical methods, exponential integrator, ordinary differential equations (ODEs), partitioned methods, Rush–Larsen method, simulation of electrophysiological models, stiffness.**

I. Introduction

ACCORDING to the World Health Organization, ischaemic heart disease was the single leading cause of death overall in its member countries in 2008 [1]. Many heart problems can be linked to abnormalities in the electrical activity in the heart.

The electrophysiological behavior of the heart can be mathematically modeled by differential equations. In particular, the electrical activity and ionic currents of a single heart cell can be described by a system of ordinary differential equations (ODEs). These ODEs are coupled with a system of partial differential equations (PDEs) in order to model the propagation of the electrical activity throughout the entire heart via the monodomain or bidomain model [2].

A common way to solve the monodomain or bidomain model is via operator splitting, an algorithm that generally splits a system of differential equations into a number of subsystems and solves each separately. In particular, when solving the monodomain or bidomain model numerically, a system of ODEs for the averaged electrical activity of a number of myocardial cells must be solved at each node of the discretized spatial domain. Accordingly, the efficiency of the numerical method used for solving the ODEs for cell models plays an important role in solving the monodomain or bidomain model efficiently.

In this study, we consider the numerical solution of 37 verified myocardial cell models from the CellML model repository [3]; see also [4] and [5]. The range of cell models encompasses widely varying degrees of stiffness that can be characterized by analyzing the eigenvalues of the Jacobian matrix [4]. The level of stiffness of a particular model determines whether a numerical method can solve the model efficiently.

One of the most popular methods for solving the ODEs that describe the dynamic behavior of myocardial cell models is known as the Rush–Larsen (RL) method [6]. Its popularity stems from its improved stability properties over integrators such as the forward Euler (FE) method coupled with its easy implementation. There have been recent attempts to build on the success of RL [7], [8]. In particular, a generalized RL method of second order, which we denote by GRL2, was proposed in [7], where it was shown to outperform RL on three cell models. A generalized RL method of first order, which we denote by GRL1, was also described but not investigated. GRL2 only outperformed the explicit midpoint rule, a standard second-order explicit Runge–Kutta method, on the single stiff cell model used in the study. This qualitative characterization of performance of GRL2 relative to RL and FE is confirmed in [4] (see also [5]) on 37 cell models, i.e., RL is the most efficient method on the majority of cell models, with GRL2 being most successful on the stiffest ones. This also leads to the observation that most cell models are only moderately stiff, lending a way to understand the success of RL as an efficient general-purpose method.

The RL method is a *partitioned* method [9] for solving ODEs. It partitions the ODEs into two sets: one for the gating variables, which are treated by an exponential integrator, and another for the remaining equations, which are treated by the FE method.

The earliest publications on exponential integration date back as far as [10] and [11]. Besides cardiac electrophysiology, exponential integration is also popular in neural simulations, e.g., [12], [13], mainly in the context of linear, constant-coefficient ODEs. With the advent of efficient algorithms for evaluating

Manuscript received January 8, 2012; revised April 26, 2012; accepted May 25, 2012. Date of publication June 21, 2012; date of current version August 16, 2012. This work was supported by the National Science and Engineering Research Council of Canada, the Mprime Canadian Network of Centers of Excellence, and the Simula Research Laboratory, Norway. *Asterisk indicates corresponding author.*

M. E. Marsh is with Solido Design Automation, Inc., Saskatoon, SK, S7N 3R3, Canada (e-mail: mel864@mail.usask.ca).

S. T. Ziaratgahi is with the Department of Mathematics and Statistics, University of Saskatchewan, Saskatoon, Saskatchewan S7N 5E6, Canada (e-mail: saeed.torabi@usask.ca).

*R. J. Spiteri is with the Department of Computer Science, University of Saskatchewan, Saskatoon, Saskatchewan S7N 5A9, Canada (e-mail: spiteri@cs.usask.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TBME.2012.2205575

products of matrix exponentials with vectors, there has been a relatively recent resurgence in interest in exponential integration in other applications, in particular problems where the eigenvalues of the Jacobian of the solution have large negative real parts (such as for spatial discretizations of parabolic problems) or are large and purely imaginary (such as in highly oscillatory problems), e.g., see [14] for an in-depth survey. The exponential method used as part of the RL method corresponds to the exponential Euler method [14] with the Jacobian matrix approximated by its diagonal. Interestingly, this appears to be the sense in which exponential integration was proposed in [11]; however, it falls short of the modern definition of the method.

The success of the RL method can be understood in terms of the relatively good stability provided by the exponential part of the integration when treating the gating variables. However, this approach may not be expected to work well on cell models for which the stiffness is not captured by the gating equations.

In this study, we focus on the performance of three basic numerical methods, FE, RL, and GRL1. All of these methods are first order. The order of a method has an effect on the amount of computation required for the method to reach a given level of accuracy. However, for the purposes of this study, we wish to remove any potentially confounding issues of order and focus only on issues of stability and stiffness. By analyzing the eigenvalues of the Jacobian matrix of the stiff cell models, we find that only a few of the equations are responsible for the stiffness, and in many cases, these equations are not associated with gating variables. In addition, the ODEs classified as stiff are in fact not stiff on the entire interval of integration. Using this information, we construct a partitioned method combining GRL1 with FE that handily outperforms the three basic methods on five stiff models.

The remainder of this paper is organized as follows. In Section II, we give a mathematical description of myocardial cell models and briefly discuss stiffness for ODEs. In Section III, we review the three numerical methods considered and introduce the partitioned method GRL1/FE|FE that combines GRL1 with FE. In Section IV, we present the numerical experiments and assess the performance of each method for solving cell models. In order to do this, we introduce a new error norm to assess the accuracy of numerical solutions of cell models. Finally, in Section V, we summarize our conclusions.

II. Myocardial Cell Models

A wide range of models has been developed to describe the electrical currents in various single heart cells, e.g., atrial cells, ventricular cells, human cells, rat cells, etc. Most can be formulated as an initial-value problem (IVP) for a system of ODEs of the form

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(t, \mathbf{y}), \qquad \mathbf{y}(t_0) = \mathbf{y_0}. \tag{1}$$

The component variables of the vector $\mathbf{y}$ are dependent on the cell model but they typically include the transmembrane potential, a number of gating variables, and a set of ionic concentrations. Many important cell models are derived from the Hodgkin–Huxley model of a squid giant axon, first proposed in 1952 [15]. This type of model can be written as

$$\frac{dV_m}{dt} = -\frac{1}{C_m}\sum_{i=1}^{n_{\text{ion}}} I_i(V_m, \mathbf{m}, \mathbf{c}, t), \tag{2a}$$

$$\frac{dc_j}{dt} = g_j(c_j, \mathbf{m}, V_m, t), \qquad j = 1, 2, \ldots, n_c, \tag{2b}$$

$$\frac{dm_k}{dt} = \alpha_k(1 - m_k) - \beta_k m_k, \qquad k = 1, 2, \ldots, n_m. \tag{2c}$$

Equation (2a) describes the evolution of the transmembrane potential V_m, where I_i is the total transmembrane current carried by ion i of n_{ion} ions and C_m is the capacitance of the cell membrane per unit area. Equation (2b) describes the dynamic variations in n_c intracellular ionic concentrations. Equation (2c) describes the opening and closing of n_m ion channels in the cell membrane expressed by the gating variable vector $\mathbf{m}$ with components m_k, where $\alpha_k = \alpha_k(V_m)$ and $\beta_k = \beta_k(V_m)$. The ODEs given by (2a) and (2b) are generally nonlinear; however, α_k and β_k in (2c) are only nonlinear functions of V_m.

In this study, we consider 37 verified myocardial cell models from the CellML model repository [3]; see also [4] and [5]. Table I contains the name of each model, the reference to the original paper, the total number of variables, the number of gating variables, and a brief description of the model. We note that the model of Winslow *et al.* [47] used in this study has 31 variables, representing a reduced form of the original model, and is subsequently referred to as Winslow31. In Winslow31, the intracellular sodium concentration and one of the calcium handling mechanisms from the original model are taken as constants [16].

An important consideration in the efficient numerical solution of differential equations is the concept of *stiffness*. Despite its pervasiveness in practice, there is no universally accepted theoretical definition of stiffness. In this study, an IVP (1) is considered to be stiff on a time interval with respect to a given numerical method and error tolerance when stability requirements force the numerical method to take smaller step sizes than those dictated by accuracy requirements [49]. Generally, step sizes required for a non-stiff method applied to a stiff model are much smaller than accuracy requirements dictate, resulting in a numerical solution that is much more accurate (and hence more costly to compute) than desired. In order to reduce computational effort, it is preferred that step sizes be chosen based only on accuracy requirements.

The cell models considered in this paper range from non-stiff to moderately stiff to stiff for typical accuracy requirements. The level of stiffness of a particular model determines whether a given numerical method can solve the model efficiently. The characterization of stiffness in each cell model is, therefore, important in order to choose the appropriate numerical method to efficiently solve that particular model to a given accuracy. Given the wide range of cell models and their associated levels of stiffness, it is not surprising that no single numerical method is the most effective on all the models.

Related to the stiffness of an IVP (1) are the eigenvalues of the Jacobian matrix $\mathbf{J} = \frac{\partial \mathbf{f}}{\partial \mathbf{y}}(t, \mathbf{y})$ evaluated over time. The magnitude and nature of these eigenvalues (i.e., whether they are real, imaginary, or complex) can provide information as to

TABLE I
SUMMARY OF THE 37 MYOCARDIAL CELL MODELS STUDIED

Model	Reference	Number of variables	Number of gating variables	Description
Beeler–Reuter (1977)	[17]	8	6	Mammalian ventricle
Bondarenko et al. (2004)	[18]	41	8	Mouse ventricle
Courtemanche et al. (1998)	[19]	21	15	Human atrium
Demir et al. (1994)	[20]	27	10	Rabbit sinoatrial node
Demir et al. (1999)	[21]	29	11	Rabbit sinoatrial node
DiFrancesco–Noble (1985)	[22]	16	9	Mammal Purkinje fibre
Dokos et al. (1996)	[23]	18	8	Rabbit sinoatrial node
Faber–Rudy (2000)	[24]	19	12	Guinea pig ventricle
FitzHugh–Nagumo (1961)	[25], [26]	2	0	Nerve membrane
Fox et al. (2002)	[27]	13	10	Canine ventricle
Hilgemann–Noble (1987)	[28]	15	3	Rabbit atrium
Hund–Rudy (2004)	[29]	29	20	Canine ventricle
Jafri et al. (1998)	[30]	31	5	Guinea pig ventricle
Luo–Rudy (1991)	[31]	8	6	Guinea pig ventricle
Maleckar et al. (2008)	[32]	30	12	Human atrium
McAllister et al. (1975)	[33]	10	9	Canine Purkinje fibre
Noble (1962)	[34]	4	3	Mammal Purkinje fibre
Noble–Noble (1984)	[35]	15	8	Rabbit sinoatrial node
Noble et al. (1991)	[36]	17	6	Guinea pig ventricle
Noble et al. (1998)	[37]	22	8	Guinea pig ventricle
Nygren et al. (1998)	[38]	29	12	Human atrium
Pandit et al. (2001)	[39]	26	12	Rat left-ventricle
Pandit et al. (2003)	[40]	26	13	Rat left-ventricle
Puglisi–Bers (2001)	[41]	17	11	Rabbit ventricle
Sakmann et al. (2000)*	[42]	21	6	Guinea pig ventricle
Stewart et al. (2009)	[43]	20	13	Human Purkinje fibre
Ten Tusscher et al. (2004)*	[44]	17	10	Human ventricle
Ten Tusscher et al. (2006)*	[45]	19	12	Human ventricle
Wang–Sobie (2008)	[46]	35	11	Neonatal mouse ventricle
Winslow31	[47]	31	8	Canine ventricle
Zhang et al. (2000)	[48]	15	14	Rabbit sinoatrial node

Three types of myocardial cell model variants (endocardial cell, epicardial cell, and M-cell) exist for each of the models marked with an asterisk.

the degree of stiffness present in an IVP at a given time. A stiff IVP typically has eigenvalues λ with large negative real parts on some time interval. Such eigenvalues force the time step Δt to be small so that $\lambda \Delta t$ is within the stability region of the numerical method. IVPs that have eigenvalues with large imaginary parts also tend to be difficult to solve by standard solvers; however, they are not normally considered to be stiff according to the classical treatment of stiffness.

The extreme values for the real and imaginary components of the eigenvalues for the 37 cell models studied are reported in Table II along with the percentage of time when a complex eigenvalue pair was present. For the typical accuracies with which we are concerned in this study, the models with small negative real eigenvalues, such as the FitzHugh–Nagumo model, are considered to be non-stiff. Similarly, the models with large negative real eigenvalues, such as the model of Pandit *et al.* [40] and Winslow31, are considered to be stiff.

III. NUMERICAL METHODS

The solutions to myocardial cell models must generally be obtained through the use of numerical methods. One common numerical method used to solve an IVP (1) is the FE method. The FE method is a first-order explicit method that is widely used because of its ease of implementation. However, the FE method is often severely limited by stability constraints when problems are stiff.

We focus on the performance of three basic methods: FE, RL, and GRL1. Their formulation is presented in detail and they are assessed in terms of their efficiency in solving the 37 cell models listed in Table I. The FE and GRL1 methods are subsequently combined into a partitioned method that is more computationally efficient than the RL method for stiff cell models.

A. Basic Methods

Given the IVP

$$\frac{d\mathbf{y}}{dt} = \mathbf{f}(t, \mathbf{y}), \qquad \mathbf{y}(t_n) = \mathbf{y_n}, \tag{3}$$

for $t_n < t < t_{n+1}$, where $\mathbf{y} \in \mathbb{R}^M$, $\mathbf{f} : \mathbb{R} \times \mathbb{R}^M \to \mathbb{R}^M$, and $\Delta t_n = t_{n+1} - t_n$, the FE method approximates (3) by

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \Delta t_n \mathbf{f}(t_n, \mathbf{y}_n). \tag{4}$$

The RL method applies the FE method to the ODEs for non-gating variables present in (3) but uses a different technique for the ODEs satisfied by gating variables. These ODEs have the form of (2c) that, for a typical gating variable y, can be reformulated as

$$\frac{dy}{dt} = \frac{y_\infty - y}{\tau_y}, \tag{5}$$

TABLE II
EXTREME VALUES OF THE EIGENVALUES OF THE JACOBIAN FOR EACH CELL MODEL

Model	$\min(Re(\lambda))$	$\max(Re(\lambda))$	$\min(Im(\lambda))$	$\max(Im(\lambda))$	% Complex
Beeler–Reuter (1977)	–8.20E+1	1.55E–2	–1.97E+0	1.97E+0	45
Bondarenko et al. (2004)	–8.49E+3	4.51E+0	–2.80E+0	2.80E+0	53
Courtemanche et al. (1998)	–1.29E+2	1.87E–1	–4.50E+0	4.50E+0	82
Demir et al. (1994)	–3.80E+1	4.79E–1	–7.95E–2	7.95E–2	74
Demir et al. (1999)	–3.82E+1	4.81E–1	–7.95E–2	7.95E–2	72
DiFrancesco–Noble (1985)	–2.63E+1	1.88E+0	–6.14E–1	6.14E–1	56
Dokos et al. (1996)	–2.99E+1	5.06E–1	–1.19E–1	1.19E–1	97
Faber–Rudy (2000)	–1.84E+2	1.37E–2	–5.61E–1	5.61E–1	58
FitzHugh–Nagumo (1961)	–4.39E–1	1.78E–1	–4.59E–2	4.59E–2	28
Fox et al. (2002)	–4.39E+2	4.44E–2	–4.19E–1	4.19E–1	65
Hilgemann–Noble (1987)	–3.25E+1	1.58E–1	–2.25E–1	2.25E–1	25
Hund–Rudy (2004)	–1.95E+2	9.22E–1	–3.74E+0	3.74E+0	62
Jafri et al. (1998)	–4.42E+3	4.82E+0	–2.35E–1	2.35E–1	47
Luo–Rudy (1991)	–1.51E+2	7.01E–2	–4.11E–2	4.11E–2	73
Maleckar et al. (2008)	–4.16E+1	2.42E–1	–3.43E–1	3.43E–1	28
McAllister et al. (1975)	–1.83E+2	1.49E+0	–3.02E+0	3.02E+0	68
Noble (1962)	–9.80E+0	1.74E+0	–1.28E–1	1.28E–1	24
Noble–Noble (1984)	–1.25E+1	4.77E–1	–1.03E–1	1.03E–1	92
Noble et al. (1991)	–3.89E+1	4.35E+0	–1.72E–1	1.72E–1	20
Noble et al. (1998)	–3.60E+1	5.71E+0	–2.35E–1	2.35E–1	47
Nygren et al. (1998)	–4.03E+1	2.05E+0	–3.88E–1	3.88E–1	24
Pandit et al. (2001)	–6.92E+3	4.30E+0	–1.43E+0	1.43E+0	12
Pandit et al. (2003)	–7.54E+4	3.87E+0	–9.11E–1	9.11E–1	35
Puglisi–Bers (2001)	–1.91E+2	2.22E+0	–1.07E–1	1.07E–1	41
Sakmann et al. (2000) – Endo	–2.97E+1	7.21E–1	–7.48E–2	7.48E–2	84
Sakmann et al. (2000) – Epi	–2.96E+1	6.98E–1	–7.47E–2	7.47E–2	75
Sakmann et al. (2000) – M-cell	–2.98E+1	1.98E+0	–7.58E–2	7.58E–2	72
Stewart et al. (2009)	–1.38E–1	3.34E–3	–1.57E–3	1.57E–3	92
Ten Tusscher et al. (2004) – Endo	–1.17E+3	1.01E–1	–4.64E+0	4.64E+0	17
Ten Tusscher et al. (2004) – Epi	–1.17E+3	9.74E–2	–4.70E+0	4.70E+0	18
Ten Tusscher et al. (2004) – M-cell	–1.17E+3	9.75E–2	–4.70E+0	4.70E+0	21
Ten Tusscher et al. (2006) – Endo	–1.26E+3	4.00E+0	–4.77E+0	–4.77E+0	50
Ten Tusscher et al. (2006) – Epi	–9.44E+2	2.84E+0	–5.01E+0	5.01E+0	51
Ten Tusscher et al. (2006) – M-cell	–9.81E+2	4.36E+0	–4.64E+0	4.64E+0	34
Wang–Sobie (2008)	–1.23E+2	1.23E+0	–1.24E+0	1.24E+0	46
Winslow31	–1.84E+4	1.53E+0	–4.22E–1	4.22E–1	63
Zhang et al. (2000)	–2.22E+1	1.29E–1	–1.00E–1	1.00E–1	89

The minimum real part of the set of eigenvalues is denoted min(Re(λ)) and the maximum real part of the set of eigenvalues is denoted max (Re(λ)). Similarly, the minimum and maximum imaginary parts are denoted min(Im(λ)) and max(Im(λ)). The percentage of the solution interval in which there is at least one pair of complex eigenvalues is also reported.

where

$$y_\infty = \frac{\alpha_y}{\alpha_y + \beta_y}, \qquad \tau_y = \frac{1}{\alpha_y + \beta_y},$$

and where $\alpha_y = \alpha_y(V_m)$ and $\beta_y = \beta_y(V_m)$. The RL method assumes the transmembrane potential V_m is constant over each step, allowing (5) to be treated as a linear ODE with an exact solution given by

$$y_n = y_\infty + (y_{n-1} - y_\infty) e^{-\frac{\Delta t_n}{\tau_y}}. \tag{6}$$

The GRL1 method decouples and linearizes the ODE system around a point $\mathbf{y} = \mathbf{y}_n$ at time $t = t_n$ to obtain

$$\frac{dy_i}{dt} = f_i(\mathbf{y}_n) + \frac{\partial}{\partial y_i} f_i(\mathbf{y}_n)\,(y_i - y_{n,i}), \quad y_i(t_n) = y_{n,i} \tag{7}$$

for $i = 1, 2, \ldots, M$, where the subscript i denotes component i of a vector. The exact solution of (7) is given by

$$y_i(t) = y_{n,i} + \frac{a}{b}\left(e^{b(t-t_n)} - 1\right), \quad i = 1, 2, \ldots, M, \tag{8}$$

where $a = f_i(\mathbf{y}_n)$ and $b = \partial f_i(\mathbf{y}_n)/\partial y_i$. The numerical solution $\mathbf{y}_{n+1}$ at time $t = t_{n+1}$ is obtained by

$$y_{n+1,i} = y_{n,i} + \frac{a}{b}(e^{b(\Delta t_n)} - 1), \quad i = 1, 2, \ldots, M. \tag{9}$$

In practice, if $|\partial f_i(\mathbf{y})/\partial y_i| < \delta$, where $\delta = 10^{-8}$ for double-precision calculations, the limit as $\partial f_i(\mathbf{y})/\partial y_i \to 0$ is used instead of (8) to get

$$y_i(t) = y_{n,i} + a(t - t_n), \quad i = 1, 2, \ldots, M.$$

The numerical solution, which is also exact when $\partial f_i(\mathbf{y})/\partial y_i = 0$, is then obtained by

$$y_{n+1,i} = y_{n,i} + a\Delta t_n, \quad i = 1, 2, \ldots, M.$$

In order to use the GRL1 method, the diagonal of the Jacobian matrix $\partial \mathbf{f}/\partial \mathbf{y}$ is required. Numerical Jacobians are used throughout, with a special implementation in practice because only the diagonal elements are required. This reduces computational cost because unnecessary components of the Jacobian matrix are not computed. The finite-difference approximation of $\partial f_i(\mathbf{y})/\partial y_i$ is obtained by

$$\partial f_i(\mathbf{y})/\partial y_i \approx \frac{f_i(y_1, \ldots, y_{i-1}, y_i + \Delta, y_{i+1}, \ldots, y_M) - f_i(\mathbf{y})}{\Delta},$$

where $\Delta = 10^{-8}$ for double-precision calculations.

The RL and GRL1 methods treat the gating (2c) similarly. In other words, if y_i is a gating variable then (9) reduces to (6). The key difference between the methods is in their treatment of the non-gating variables: GRL1 applies an exponential integrator based on local linearization to non-gating variables whereas RL uses the FE method.

A summary of the three basic numerical methods used for this study is presented in Table III. We note that the method that

TABLE III
SUMMARY OF BASIC NUMERICAL METHODS

Method	Gating variables (gating equations)	Non-gating variables (non-linear equations)
FE	FE integrator (4)	FE integrator (4)
RL	Exponential integrator (9)	FE integrator (4)
GRL1	Exponential integrator (9)	Local linearization (7) + Exponential integrator (9)

is the least stable but computationally cheapest per step is the FE method and the method that is the most stable but computationally costliest per step is the GRL1 method. This tradeoff of stability for computational cost per step is typical for numerical methods to solve stiff IVPs. It is often the case that the increase in stable step size more than offsets increase in computational cost per step, leading to a less expensive computation (i.e., more efficient method) overall.

B. Partitioned Methods

By analyzing the eigenvalues of the Jacobian matrix of stiff cell models, it can be determined that only a few of the ODEs are responsible for the stiffness of the model. This provides a means by which the system of ODEs can be partitioned into stiff and non-stiff subsystems. This eigenvalue analysis also reveals on which subinterval(s) of the entire interval of integration the IVP is stiff. This permits a partitioning of the interval of integration into stiff and non-stiff subintervals.

As examples, we consider the stiff cell models of Pandit *et al.* [40] and Winslow31. The plots of the real parts of the eigenvalues of the Jacobian matrix of these two models are given in Figs. 1 and 2. The negative eigenvalues and their corresponding ODEs are labeled in the figures. The plots also show that the ODEs that capture the stiffness of the system are not stiff on the entire interval of integration. From close examination of the eigenvalues, we find that only two out of the 26 ODEs from the model of Pandit *et al.* [40] [see Fig. 1(b)] and only two out of the 31 ODEs from Winslow31 [see Fig. 2(b)] are responsible for the stiffness of the models. We also identify that the stiffness is approximately contained within the subintervals [105, 195] and [0, 40] for the model of Pandit *et al.* [40] and Winslow31, respectively.

Table IV summarizes the relevant attributes of five of the stiffest models, namely those of Bondarenko *et al.* [18], Jafri *et al.* [30], and Pandit *et al.* [40], the endocardial variant of ten Tusscher *et al.* [44], and Winslow31. The models of Pandit *et al.* [39] and the epicardial and M-cell variants of the model of ten Tusscher *et al.* [44], [45] are excluded to maximize diversity. The table reports the number of variables in each model, the number of equations in the stiff subsystem, the names of the stiff ODE variables, and the stiff and non-stiff subintervals. The notation for the stiff ODE variables has been unified for the purposes of presentation.

An important point to note from Table IV is that the majority of stiff variables identified are not gating variables. This means that most of the stiffness of these models is not captured by gating variables. The exception of note is the model of ten Tusscher *et al.* [44], which from Table II is shown to be the least stiff of the five models considered.

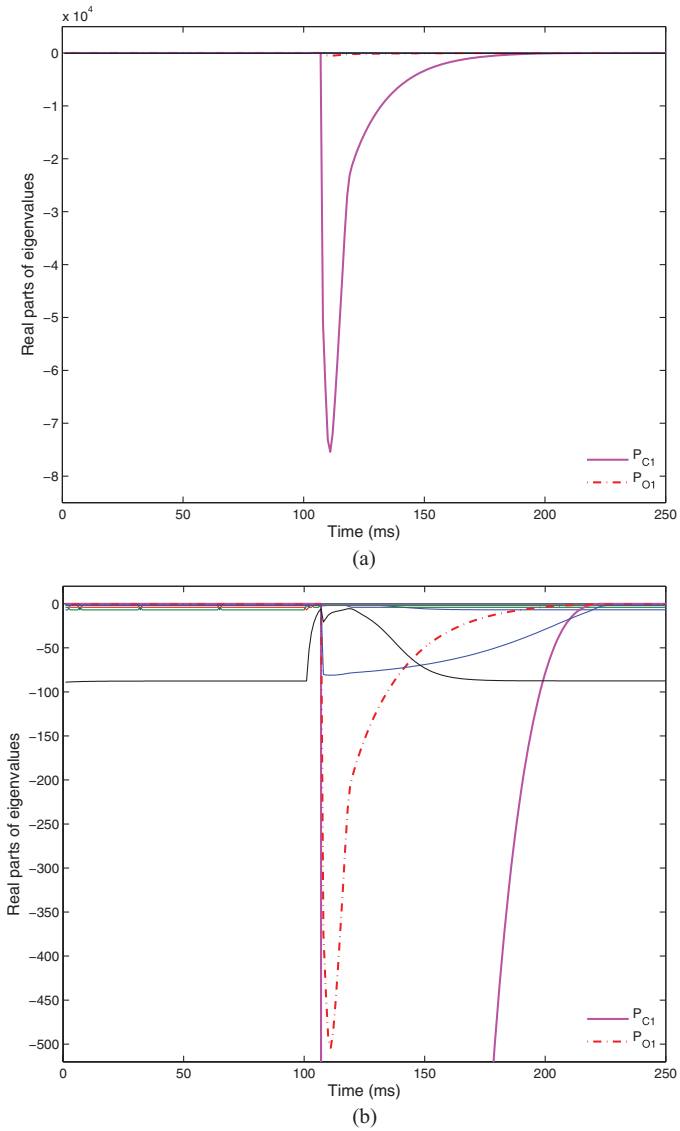

Fig. 1. Real parts of eigenvalues of Jacobian over time for the model of Pandit *et al.* [40]; stiff variables P_{C1} (—) and P_{O1} (- · - · -) are highlighted. (a) Real parts of eigenvalues of Jacobian over entire time interval. (b) Close up of real parts of eigenvalues of Jacobian over entire time interval.

We expect that the RL method is best suited for the integration of stiff models for which the stiffness is captured by the gating variables, e.g., in the model of Ten Tusscher *et al.* [44], where the only stiff variable is the gating variable m. In the case of stiff models for which the stiffness is not captured by gating variables, we expect the RL method to perform less well because its step size can be adversely impacted by stability restrictions imposed by the FE method being applied to stiff non-gating equations. For such models, we expect a method such as GRL1 that treats stiff non-gating equations with an exponential integration method to outperform the RL method. Furthermore, we expect a combination of the GRL1 method and the FE method that takes advantage of partitioning the ODE system and time

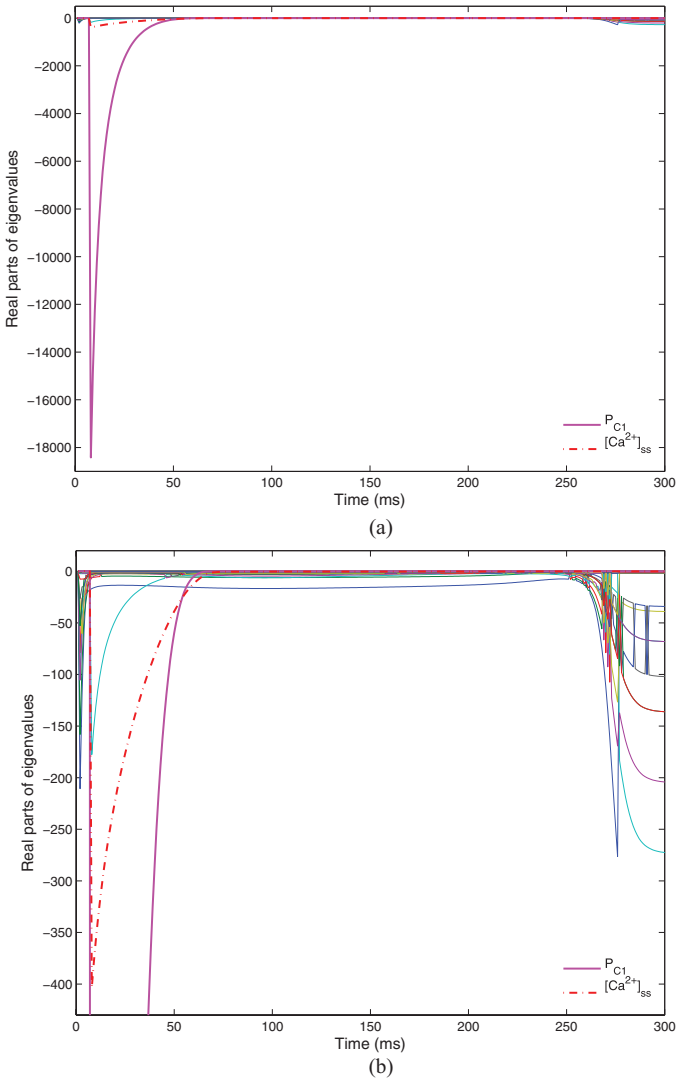

Fig. 2. Real parts of eigenvalues of Jacobian of the model Winslow31; stiff variables P_{C1} (—·) and $[Ca^{2+}]_{ss}$ (·– –·) are highlighted. (a) Real parts of eigenvalues of Jacobian over entire time interval. (b) Close up of real parts of eigenvalues of Jacobian over entire time interval.

interval into stiff and non-stiff subsets to perform even more effectively. Specifically, we propose to use the FE method for the entire ODE system on the non-stiff portion of the time domain and the GRL1 method for the stiff variables combined with the FE method for the non-stiff variables on the stiff subinterval of integration. We refer to this new partitioned method as GRL1/FE|FE.

IV. Numerical Experiments

In this section, we begin by introducing an error norm that is used to quantify the accuracy of a numerical solution of a cell model. We assess its performance in terms of reliability compared to another commonly used error norm. We then assess the performance of the three basic numerical methods FE, RL, and GRL1 on the 37 cell models listed in Table I. In the spirit of work-precision analysis, e.g., [9], the performance of a method is measured in relation to the least amount of CPU time required to achieve a specified error tolerance. Finally, we assess the performance of the proposed GRL1/FE|FE method on the five stiffest cell models from Table I.

A. Mixed Root-Mean-Square Error Norm

In order to evaluate the accuracy and efficiency of numerical methods for solving a myocardial cell model (2) over the interval $t \in [t_0, t_f]$, it is necessary to have a measure of the accuracy of the numerical solution. This is normally done by computing an average of the error in the transmembrane potential V_m at N points in $t \in [t_0, t_f]$. However, in order to compute an average of the error, either the exact solution must be known or a reference solution must be computed for all N points. In this case, a reference solution is a numerical solution to (2) that is known to have converged to d digits of accuracy in V_m at all N points, where d is sufficiently large and determined by comparing increasingly accurate solutions and counting the number of matching digits for all N points.

A error norm commonly used to assess the accuracy of the numerical solution of a myocardial cell model is the relative root mean square (RRMS) error, defined by

$$e_{\mathrm{RRMS}} = \sqrt{\frac{1}{N}\frac{\sum_{i=1}^{N}(\hat{V}_{m,i} - V_{m,i})^2}{\sum_{i=1}^{N}\hat{V}_{m,i}^2}}$$

where $V_{m,i}$ is the numerical solution and $\hat{V}_{m,i}$ is the reference solution, both at time t_i. We introduce a new error norm, which we call the mixed root mean square (MRMS) error, defined by

$$e_{\mathrm{MRMS}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{\hat{V}_{m,i} - V_{m,i}}{1 + |\hat{V}_{m,i}|}\right)^2}.$$

The numerical solutions produced by using the RRMS and MRMS error norms at 5% and 1% error are compared for the model of McAllister *et al.* [33], solved using the RL method. Fig. 3 compares a reference solution to numerical solutions computed at 5% and 1% RRMS error for the transmembrane potential. It can be seen that at 5% RRMS error, the action potential is early by approximately 100 ms. At 1% RRMS error, the agreement is clearly much better. On the scale of Fig. 3, the numerical solutions produced at 5% and 1% MRMS error are in close agreement with the reference solution. Fig. 4 compares a reference solution to solutions computed to 5% and 1% MRMS error for the transmembrane potential for the subinterval [200, 250] ms. It can be seen that at 1% MRMS error, the numerical solution is extremely accurate. At 5% MRMS error, the numerical solution perhaps remains sufficiently accurate but is eight times faster to compute.

From examination of Figs. 3 and 4, we postulate that the MRMS error norm at 5% strikes an appropriate balance for clinical accuracy requirements while taking into account computational effort. We do not generally advocate the use of the RRMS error norm because it is sensitive to the error level specified; i.e., disproportionately inaccurate solutions can satisfy the RRMS error norm for seemingly reasonable levels. Because it is a norm, it can safely be used at low error levels. Similar observations can be made for the RRMS and MRMS error norms

TABLE IV
STIFF MODELS AND THEIR ATTRIBUTES

Model	Size	No. of stiff ODEs	Stiff ODE variables	Stiff subinterval	Non-stiff subinterval
Bondarenko et al. (2004)	41	2	P_{O1}, $[Ca^{2+}]_{ss}$	$[20, 75]$	$[0, 20]$
Jafri et al. (1998)	31	4	P_{C1}, $[Ca^{2+}]_{ss}$, C_{Ca0}, C_{Ca1}	$[0, 50]$	$[50, 300]$
Pandit et al. (2003)	26	2	P_{O1}, P_{C1}	$[105, 195]$	$[0, 105] \cup [195, 250]$
Ten Tusscher et al. (2004) – Endo	19	1	m	$[0, 12] \cup [290, 400]$	$[12, 290]$
Winslow31	31	2	P_{C1}, $[Ca^{2+}]_{ss}$	$[0, 40]$	$[40, 300]$

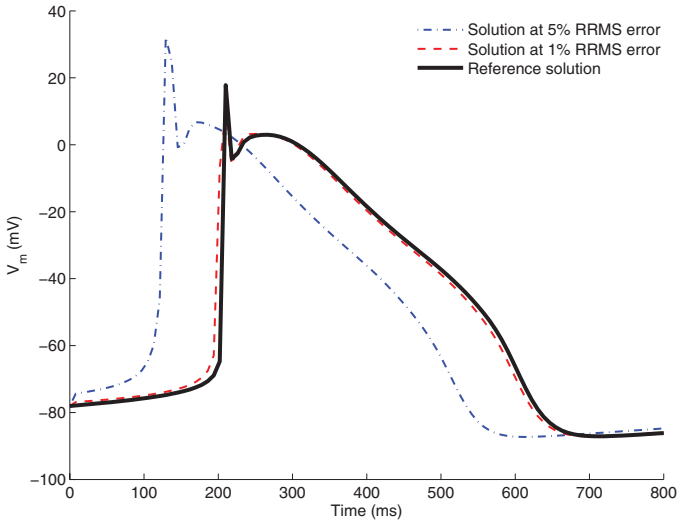

Fig. 3. Reference solution (——) and numerical solutions for V_m produced by the RL method at RRMS errors of 5% (· – · – ·) and 1% (– – –) for the model of McAllister *et al.* [33].

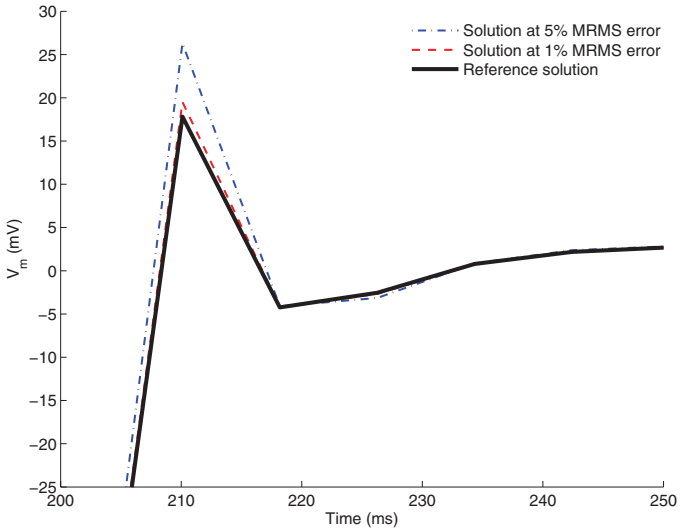

Fig. 4. Close up of reference solution (——) and numerical solutions for V_m produced by the RL method at MRMS errors of 5% (· – · – ·) and 1% (– – –) for the model of McAllister *et al.* [33] on the interval [200, 250].

for the remaining cell models. Accordingly, only results for the MRMS error norm at 5% are presented.

B. Simulation Results

The results from solving the 37 cell models with FE, RL, and GRL1 are listed in Table V. MATLAB's `ode15s` method [50] was used with a sequence of decreasing absolute and relative tolerances to 10^{-12} to compute reference solutions with seven to ten matching digits at $N = 100$ equally spaced points in the intervals of integration. The MRMS error between the reference solution and the computed solution was computed using linear interpolation as necessary. A significant part of our analysis consisted of determining the maximum constant step sizes that satisfied a 5% MRMS error tolerance for each of the models with respect to the reference solutions. This enabled us to determine the efficiency of a numerical method as the amount of computation time (i.e., the product of the number of steps and the computational cost per step) required to achieve a given accuracy (see, e.g., [9]). Timings reported are the minimum run time out of 100 runs for these step sizes. Constant step sizes are used to reflect the scenario of the ODEs being solved within the context of solving the monodomain or bidomain model via operator splitting. Timings were computed in MATLAB R2010a on an HP Z400 with an Intel Xeon W3520 2.66 GHz quad-core processor with 16 GB of DDR3 RAM running 64-bit Ubuntu 9.04. Hyperthreading and turbo-boost were enabled while the timings were computed.

From Table V, we find that the FE method wins on nine models, the RL method wins on 24 models, and the GRL1 method wins on four models. Moreover, the RL method is never more than about 50% less efficient than the FE method. This confirms that the popularity of the RL method in practice is well justified. The secrets to its success lie mainly in its partitioning of the ODE system into gating and non-gating variables and solving the equations for the gating variables with an exponential integrator. The RL method has the best combination of stability and computational expense per step for moderately stiff models. Because the majority of the 37 cell models are moderately stiff, the RL method is the best single method for most models. The GRL1 method is the most efficient for three of the stiffest models, those of Bondarenko *et al.* [18], Pandit *et al.* [39], and Pandit *et al.* [40].

The performance of the partitioned method GRL1/FE|FE is determined for five of the stiffest cell models, namely those of Bondarenko *et al.* [18], Jafri *et al.* [30], Pandit *et al.* [40], Ten Tusscher *et al.* [45], and Winslow31, according to the partitions reported in Table IV, and the results are reported in Table VI. For GRL1/FE|FE, Δt_{ns} and Δt_s are the step sizes used in the non-stiff and stiff regions, respectively.

From Table VI, we see that GRL1/FE|FE is the most efficient method for all five of the stiff models considered. For the two stiffest models, namely that of Pandit *et al.* [40] and Winslow31, GRL1/FE|FE is almost five and three times faster, respectively, than its next closest competitor. For these models, RL is not the most efficient basic method; GRL1/FE|FE is about 270 and 3

TABLE V
STEP SIZE, IN MILLISECONDS, AND EXECUTION TIME, IN SECONDS, OF THE FOUR NUMERICAL METHODS USING THE LARGEST STEP SIZE THAT GIVES LESS THAN 5% MRMS ERROR

Model	FE		RL		GRL1	
	Δt	Time	Δt	Time	Δt	Time
Beeler–Reuler (1977)	2.53E–2	3.53E–2	7.20E–1	**1.40E–3**	8.08E–1	3.85E–3
Bondarenko et al. (2004)	2.13E–4	2.23E+0	2.13E–4	2.28E+0	7.47E–3	**8.41E–1**
Courtemanche et al. (1998)	1.94E–2	2.11E–1	7.97E–2	**5.60E–2**	9.60E–2	3.01E–1
Demir et al. (1994)	5.95E–2	**1.52E–2**	5.32E–2	1.76E–2	1.18E–1	9.03E–2
Demir et al. (1999)	5.96E–2	**1.74E–2**	4.73E–2	2.36E–2	9.99E–2	1.26E–1
DiFrancesco–Noble (1985)	7.73E–2	8.21E–2	1.95E–1	**3.40E–2**	2.07E–1	3.22E–1
Dokos et al. (1996)	7.02E–2	2.87E–2	1.22E–1	**1.64E–2**	8.02E–2	2.78E–1
FitzHugh–Nagumo (1961)	2.72E–3	**6.02E–3**	NA	NA	2.60E–3	1.35E–1
Faber–Rudy (2000)	1.12E–2	2.05E–1	2.01E–2	**1.17E–1**	4.06E–2	6.28E–1
Fox et al. (2002)	4.62E–3	2.94E–1	4.33E–2	**3.31E–2**	1.16E–1	8.77E–2
Hilgemann–Noble (1987)	6.25E–2	1.93E–2	8.06E–2	**1.51E–2**	1.52E–1	9.77E–2
Hund–Rudy (2004)	7.80E–3	**3.11E–1**	5.33E–3	4.85E–1	5.47E–3	4.88E+0
Jafri et al. (1998)	5.76E–4	3.65E+0	5.77E–4	**3.59E+0**	1.41E–3	1.71E+1
Luo–Rudy (1991)	1.35E–2	1.33E–1	1.23E–1	1.30E–2	3.15E–1	**1.01E–2**
Maleckar et al. (2008)	5.02E–2	7.86E–2	8.87E–2	**4.60E–2**	4.20E–1	1.29E–1
McAllister et al. (1975)	2.47E–2	7.35E–2	4.69E–1	**4.41E–3**	2.53E–1	2.38E–2
Noble (1962)	2.02E–1	**2.83E–3**	1.47E–1	3.69E–3	1.10E–1	1.77E–2
Noble–Noble (1984)	2.04E–1	**5.65E–3**	1.21E–1	9.57E–3	9.27E–2	1.21E–1
Noble et al. (1991)	5.15E–2	2.15E–2	1.53E–1	**7.46E–3**	1.04E–1	1.17E–1
Noble et al. (1998)	5.56E–2	5.37E–2	1.57E–1	**1.96E–2**	8.86E–2	3.47E–1
Nygren et al. (1998)	5.36E–2	9.44E–2	8.88E–2	**5.88E–2**	2.06E–1	2.77E–1
Pandit et al. (2001)	2.91E–4	4.94E+0	2.91E–4	5.13E+0	2.40E–2	**6.02E–1**
Pandit et al. (2003)	2.65E–5	5.55E+1	2.65E–5	5.68E+1	1.57E–2	**9.67E–1**
Puglisi–Bers (2001)	5.97E–3	1.39E+0	1.45E–2	**7.81E–1**	3.23E–2	1.04E+0
Sakmann et al. (2000) – Endo	6.90E–2	**5.12E–2**	4.99E–2	6.94E–2	4.16E–2	8.87E–1
Sakmann et al. (2000) – Epi	6.90E–2	**5.24E–2**	4.16E–2	8.32E–2	3.83E–2	9.67E–1
Sakmann et al. (2000) – M-cell	6.86E–2	5.26E–2	2.32E–1	**1.51E–2**	4.21E–1	8.80E–2
Stewart et al. (2009)	1.52E+1	4.42E–1	2.05E+2	**3.48E–2**	1.74E+2	3.78E–1
Ten Tusscher et al. (2004) –Endo	1.78E–3	1.77E+0	1.24E–1	**2.65E–2**	1.37E–1	2.18E–1
Ten Tusscher et al. (2006) –Endo	1.62E–3	1.29E+0	7.03E–2	**3.10E–2**	1.29E–1	1.67E–1
Ten Tusscher et al. (2004) –Epi	1.78E–3	1.79E+0	1.12E–1	**2.97E–2**	1.19E–1	2.51E–1
Ten Tusscher et al. (2006) –Epi	2.14E–3	9.86E–1	1.16E–1	**1.90E–2**	1.75E–1	1.23E–1
Ten Tusscher et al. (2004) –M-cell	1.76E–3	1.33E+0	1.21E–1	**2.03E–2**	1.02E–1	2.23E–1
Ten Tusscher et al. (2006) –M-cell	2.06E–3	1.01E+0	1.27E–1	**1.72E–2**	1.38E–1	1.54E–1
Wang–Sobie (2008)	1.66E–2	6.21E–2	5.27E–2	**1.90E–2**	9.36E–2	1.20E–1
Winslow31	1.07E–4	**1.41E+1**	1.07E–4	1.49E+1	9.38E–5	2.15E+2
Zhang et al. (2000)	9.97E–2	5.14E–2	4.57E–1	**1.16E–2**	3.04E–1	1.12E–1

The shortest execution time has been highlighted in bold text for each model.

TABLE VI
STEP SIZE, IN MILLISECONDS, AND EXECUTION TIME, IN SECONDS, OF THE BASIC METHODS (FE, RL, AND GRL1) AND THE PARTITIONED METHOD GRL1/FE|FE USING THE LARGEST STEP SIZES THAT GIVES LESS THAN 5% MRMS ERROR FOR FIVE OF THE STIFFEST CELL MODELS

Model	FE		RL		GRL1		GRL1/FE|FE		
	Δt	Time	Δt	Time	Δt	Time	Δt_{ns}	Δt_s	Time
Bondarenko et al. (2004)	2.13E–4	2.23E+0	2.13E–4	2.28E+0	7.47E–3	8.41E–1	3.02E–2	7.59E–3	**9.59E–2**
Jafri et al. (1998)	5.76E–4	3.65E+0	5.77E–4	3.59E+0	1.41E–3	1.71E+1	5.29E–3	2.29E–3	**6.84E–1**
Pandit et al. (2003)	2.65E–5	5.55E+1	2.65E–5	5.68E+1	1.57E–2	9.67E–1	1.99E–2	6.59E–3	**2.10E–1**
Ten Tusscher et al. (2006) –Endo	1.62E–3	1.29E+0	7.03E–2	3.10E–2	1.29E–1	1.67E–1	1.08E–1	1.00E–1	**2.55E–2**
Winslow31	1.07E–4	1.41E+1	1.07E–4	1.49E+1	9.38E–5	2.15E+2	5.00E–3	7.70E–5	**4.85E+0**

The shortest execution time has been highlighted in bold text for each model. For GRL1/FE|FE, Δt_{ns} is the step size used by the FE method in the non-stiff region and Δt_s is the stiff step size used by the GRL1/FE method in the stiff region.

times faster than RL in these cases. For the slightly less stiff models of Bondarenko *et al.* [18] and Jafri *et al.* [30], GRL1/FE|FE is about nine and five times faster than its next closest competitor, respectively. We note RL is the most efficient basic method for the model of Jafri *et al.* [30]. Finally, for the endocardial variant of the model of Ten Tusscher *et al.* [45] that only has a single stiff variable, the gating variable m, GRL1/FE|FE is 12% faster than its next closest competitor, RL.

These improvements can generally be understood as follows. In the stiff regions, the GRL1/FE|FE method can generally take larger step sizes than the RL method applied to the entire region because the partitioning of the ODEs better captures the stiffness for treatment by the exponential integrator than partitioning along the lines of gating versus non-gating variables. Moreover, because the use of GRL1 is limited to the relatively small number of stiff ODEs, it is also computationally cheaper per step than RL. In the non-stiff regions, GRL1/FE|FE reduces to FE, which is the cheapest method per step out of those considered. The GRL1/FE|FE method can also generally take larger steps on these regions than the corresponding steps for FE applied to the entire region because it is not impacted by restrictions from the stiff regions. Based on this analysis, we also expect that a method

based on partitioning the time interval into stiff and non-stiff regions and applying RL and FE, respectively, to these regions would outperform RL (see [4] and [5]) but not GRL1/FE|FE.

When solving the monodomain or bidomain model in practice, other devices, such as table lookups of activation and inactivation variables and switching to larger time steps when all model points are depolarized, are often used to reduce computation times [51]. The use of such devices would not be expected to alter the applicability of the results of this study to the monodomain or bidomain model. First, assuming the use of table lookups is used on the dominant computational part of $\mathbf{f}(t, \mathbf{y})$ from (1), its use would align the computational expense per step for methods that use RL even more closely with that of FE. This would increase the competitiveness of such RL-based methods, especially for large simulations. Nonetheless, the relative times between methods should remain comparable to those reported. Second, switching to larger time steps when all model points are depolarized can be related to the strategy of adjusting the method and time step for each point based on stiffness/non-stiffness intervals as described in Tables IV and VI. The repolarization phase roughly corresponds to the non-stiff regions. Hence, the results from Table VI can be thought of as a mechanism to increase the step size *and* switch to the most appropriate numerical method to reduce the overall computation time.

V. Conclusion

Because of its overall efficiency and relative ease of implementation, the Rush–Larsen method is a popular and effective method for solving the ODEs that describe the evolution of dynamic myocardial cell models. The Rush–Larsen method partitions the ODE system into gating and non-gating variables and solves the equations for the gating variables with an exponential integrator and the equations for the non-gating variables with the FE method. However, this approach cannot be expected to work well on cell models for which the stiffness is not captured by the gating variables. In this paper, we demonstrate that in fact the stiffness in the stiffest cell models is caused by non-gating variables, thus leading to underperformance of the Rush–Larsen method. We demonstrate that a generalized Rush–Larsen method of first order performs well on the stiffest cell models. Using an eigenvalue analysis, we are able to partition the ODEs and the interval of integration into stiff and non-stiff subsets and hence propose a partitioned method based on the generalized Rush–Larsen and FE methods that outperforms all other basic methods considered on the stiffest cell models. In order to assess the accuracy of a numerical solution to a cell model, we also propose a new error norm based on a mixed (absolute and relative) root mean square (MRMS) error that performs more satisfactorily than the RRMS error norm.

References

[1] World Health Organization, "World Health Organization: Ten leading causes of death in 2008," (Dec. 2011) [Online]. Available: http://gamapserver.who.int/gho/interactive_charts/mbd/cod_2008/graph.html

[2] L. Tung, "A bi-domain model for describing ischemic myocardial d-c potentials," Ph.D. dissertation, Dept. Electr. Eng., Comput. Sci., Massachusetts Inst. Technol, 1978.

[3] Auckland Bioengineering Institute, "The CellML project," (Nov. 2011). [Online]. Available:http://www.cellml.org/

[4] R. J. Spiteri and R. C. Dean, "Stiffness analysis of cardiac electrophysiological models," *Ann. Biomed. Eng.*, vol. 38, no. 12, pp. 3592–3604, 2010.

[5] R. J. Spiteri and R. C. Dean, "Erratum to: Stiffness analysis of cardiac electrophysiological models," *Ann. Biomed. Eng.*, vol. 40, no. 7, pp. 1622–1625, 2012.

[6] S. Rush and H. Larsen, "A practical algorithm for solving dynamic membrane equations," *IEEE Trans. Biomed. Eng.*, vol. BME-25, no. 4, pp. 389–392, Jul. 1978.

[7] J. Sundnes, R. Artebrant, O. Skavhaug, and A. Tveito, "A second-order algorithm for solving dynamic cell membrane equations," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 10, pp. 2546–2548, Oct. 2009.

[8] M. Perego and A. Veneziani, "An efficient generalization of the Rush–Larsen method for solving electro-physiology membrane equations," *Electron. Trans. Numerical Anal.*, vol. 35, pp. 234–256, 2009.

[9] E. Hairer, S. P. Nörsett, and G. Wanner, *Solving Ordinary Differential Equations I: Nonstiff Problems.*, 2nd ed., (Springer Series in Computational Mathematics). Berlin, Germany: Springer-Verlag, 1993, vol. 8.

[10] J. Hersch, "Contribution à la méthode des équations aux différences," *Z. Agnew. Math. Phys.*, vol. 9, pp. 129–180, 1958.

[11] J. Certaine, *The Solution of Ordinary Differential Equations With Large Time Constants*. New York: Wiley, 1960, pp. 128–132.

[12] R. J. Butera and M. L. McCarthy, "Analysis of real-time numerical integration methods applied to dynamic clamp experiments," *J. Neural Eng.*, vol. 1, no. 4, pp. 187–194, 2004.

[13] S. Rotter and M. Diesmann, "Exact digital simulation of time-invariant linear systems with applications to neuronal modeling," *Biol. Cybern.*, vol. 81, no. 5/6, pp. 381–402, 1999.

[14] M. Hochbruck and A. Ostermann, "Exponential integrators," *Acta. Numerica*, vol. 19, pp. 209–286, 2010.

[15] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *J. Physiol. (Lond)*, vol. 117, no. 4, pp. 500–544, 1952.

[16] J. Sundnes, G. T. Lines, and A. Tveito, "Efficient solution of ordinary differential equations modeling electrical activity in cardiac cells," *Math. Biosci.*, vol. 172, no. 2, pp. 55–72, 2001.

[17] G. W. Beeler and H. Reuter, "Reconstruction of the action potential of ventricular myocardial fibres," *J. Physiol.*, vol. 268, no. 1, pp. 177–210, 1977.

[18] V. E. Bondarenko, G. P. Szigeti, G. C. L. Bett, S.-J. Kim, and R. L. Rasmusson, "Computer model of action potential of mouse ventricular myocytes," *Amer. J. Physiol. Heart Circ. Physiol.*, vol. 287, no. 3, pp. H1378–H1403, 2004.

[19] M. Courtemanche, R. J. Ramirez, and S. Nattel, "Ionic mechanisms underlying human atrial action potential properties: Insights from a mathematical model," *Amer. J. Physiol.*, vol. 275, no. 1, pp. H301–H321, 1998.

[20] S. S. Demir, J. W. Clark, C. R. Murphey, and W. R. Giles, "A mathematical model of a rabbit sinoatrial node cell," *Amer. J. Physiol.*, vol. 266, no. 3 Pt 1, pp. C832–C852, 1994.

[21] S. S. Demir, J. W. Clark, and W. R. Giles, "Parasympathetic modulation of sinoatrial node pacemaker activity in rabbit heart: A unifying model," *Amer. J. Physiol.*, vol. 276, no. 6, Pt 2, pp. H2221–H2244, 1999.

[22] D. DiFrancesco and D. Noble, "A model of cardiac electrical activity incorporating ionic pumps and concentration changes," *Philos. Trans. R. Soc. Lond. B Biol. Sci.*, vol. 307, no. 1133, pp. 353–398, 1985.

[23] S. Dokos, B. Celler, and N. Lovell, "Ion currents underlying sinoatrial node pacemaker activity: A new single cell mathematical model," *J. Theor. Biol.*, vol. 181, no. 3, pp. 245–272, 1996.

[24] G. M. Faber and Y. Rudy, "Action potential and contractility changes in [Na(+)](i) overloaded cardiac myocytes: A simulation study," *Biophys. J.*, vol. 78, no. 5, pp. 2392–2404, 2000.

[25] R. FitzHugh, "Impulses and physiological states in theoretical models of nerve membrane," *Biophys. J.*, vol. 1, no. 6, pp. 445–466, 1961.

[26] J. Nagumo, S. Arimoto, and S. Yoshizawa, "An active pulse transmission line simulating nerve axon," in *Proc. IRE*, vol. 50, no. 10, pp. 2061–2070, 1962.

[27] J. J. Fox, J. L. McHarg, and R. F. Gilmour, Jr., "Ionic mechanism of electrical alternans," *Amer. J. Physiol. Heart Circ. Physiol.*, vol. 282, no. 2, pp. H516–H530, 2002.

[28] D. W. Hilgemann and D. Noble, "Excitation-contraction coupling and extracellular calcium transients in rabbit atrium: Reconstruction of basic cellular mechanisms," in *Proc. R. Soc. Lond. B Biol. Sci.*, 1987, vol. 230, no. 1259, pp. 163–205.

[29] T. J. Hund and Y. Rudy, "Rate dependence and regulation of action potential and calcium transient in a canine cardiac ventricular cell model," *Circulation*, vol. 110, no. 20, pp. 3168–3174, 2004.
[30] M. S. Jafri, J. J. Rice, and R. L. Winslow, "Cardiac Ca2+ dynamics: The roles of ryanodine receptor adaptation and sarcoplasmic reticulum load," *Biophys. J.*, vol. 74, no. 3, pp. 1149–1168, 1998.
[31] C. H. Luo and Y. Rudy, "A model of ventricular cardiac action potential," *Circ. Res.*, vol. 68, no. 6, pp. 1501–1526, 1991.
[32] M. M. Maleckar, J. L. Greenstein, N. A. Trayanova, and W. R. Giles, "Mathematical simulations of ligand-gated and cell-type specific effects on the action potential of human atrium," *Prog. Biophys. Mol. Biol.*, vol. 98, no. 2–3, pp. 161–170, 2008.
[33] R. E. McAllister, D. Noble, and R. W. Tsien, "Reconstruction of the electrical activity of cardiac Purkinje fibres," *J. Physiol.*, vol. 251, no. 1, pp. 1–59, 1975.
[34] D. Noble, "A modification of the Hodgkin–Huxley equations applicable to Purkinje fibre action and pace-maker potentials," *J. Physiol.*, vol. 160, pp. 317–352, 1962.
[35] D. Noble and S. J. Noble, "A model of sino-atrial node electrical activity based on a modification of the DiFrancesco–Noble (1984) equations," in *Proc. R. Soc. Lond. B Biol. Sci.*, vol. 222, no. 1228, pp. 295–304, 1984.
[36] D. Noble, S. J. Noble, G. C. Bett, Y. E. Earm, W. K. Ho, and I. K. So, "The role of sodium-calcium exchange during the cardiac action potential," *Ann. N. Y. Acad. Sci.*, vol. 639, pp. 334–353, 1991.
[37] D. Noble, A. Varghese, P. Kohl, and P. Noble, "Improved guinea-pig ventricular cell model incorporating a diadic space, IKr and IKs, and length- and tension-dependent processes," *Can. J. Cardiol.*, vol. 14, no. 1, pp. 123–134, 1998.
[38] A. Nygren, C. Fiset, L. Firek, J. W. Clark, D. S. Lindblad, R. B. Clark, and W. R. Giles, "Mathematical model of an adult human atrial cell: The role of K+ currents in repolarization," *Circ. Res.*, vol. 82, no. 1, pp. 63–81, 1998.
[39] S. V. Pandit, R. B. Clark, W. R. Giles, and S. S. Demir, "A mathematical model of action potential heterogeneity in adult rat left ventricular myocytes," *Biophys. J.*, vol. 81, no. 6, pp. 3029–3051, 2001.
[40] S. V. Pandit, W. R. Giles, and S. S. Demir, "A mathematical model of the electrophysiological alterations in rat ventricular myocytes in type-I diabetes," *Biophys. J.*, vol. 84, no. 2, Pt 1, pp. 832–841, 2003.
[41] J. L. Puglisi and D. M. Bers, "Labheart: an interactive computer model of rabbit ventricular myocyte ion channels and Ca transport," *Amer. J. Physiol. Cell Physiol.*, vol. 281, no. 6, pp. C2049–C2060, 2001.
[42] B. F. Sakmann, A. J. Spindler, S. M. Bryant, K. W. Linz, and D. Noble, "Distribution of a persistent sodium current across the ventricular wall in guinea pigs," *Circ. Res.*, vol. 87, no. 10, pp. 910–914, 2000.
[43] P. Stewart, O. V. Aslanidi, D. Noble, P. J. Noble, M. R. Boyett, and H. Zhang, "Mathematical models of the electrical action potential of Purkinje fibre cells," *Philos. Transact. A Math. Phys. Eng. Sci.*, vol. 367, no. 1896, pp. 2225–2255, 2009.
[44] K. H. W. J. ten Tusscher, D. Noble, P. J. Noble, and A. V. Panfilov, "A model for human ventricular tissue," *Amer. J. Physiol. Heart Circ. Physiol.*, vol. 286, no. 4, pp. H1573–H1589, 2004.
[45] K. H. W. J. ten Tusscher and A. V. Panfilov, "Alternans and spiral breakup in a human ventricular tissue model," *Amer. J. Physiol. Heart Circ. Physiol.*, vol. 291, no. 3, pp. H1088–H1100, 2006.
[46] L. J. Wang and E. A. Sobie, "Mathematical model of the neonatal mouse ventricular action potential," *Amer. J. Physiol. Heart. Circ. Physiol.*, vol. 294, no. 6, pp. H2565–H2575, 2008.
[47] R. L. Winslow, J. Rice, S. Jafri, E. Marbán, and B. O'Rourke, "Mechanisms of altered excitation-contraction coupling in canine tachycardia-induced heart failure—II: Model studies," *Circ Res.*, vol. 84, no. 5, pp. 571–586, 1999.
[48] H. Zhang, A. V. Holden, I. Kodama, H. Honjo, M. Lei, T. Varghese, and M. R. Boyett, "Mathematical models of action potentials in the periphery and center of the rabbit sinoatrial node," *Amer. J. Physiol. Heart Circ. Physiol.*, vol. 279, no. 1, pp. H397–H421, 2000.
[49] U. M. Ascher and L. R. Petzold, *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*. Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM), 1998.
[50] L. F. Shampine and M. W. Reichelt, "The MATLAB ODE suite," *SIAM J. Scientific Comput.*, vol. 18, no. 1, pp. 1–22, 1997.
[51] M. C. Trudel, B. Dubé, M. Potse, R. M. Gulrajani, and L. J. Leon, "Simulation of QRST integral maps with a membrane-based computer heart model employing parallel processing," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 8, pp. 1319–1329, Aug. 2004.

Megan E. Marsh received the M.Sc. degree from the Department of Mathematics and Statistics, University of Saskatchewan, Saskatoon, Canada, in 2012, where her thesis focused on developing and improving numerical methods for simulation of electrical activity in myocardial tissue.

Saeed Torabi Ziaratgahi is currently working toward the Ph.D. degree in the Department of Mathematics and Statistics, University of Saskatchewan, Saskatoon, Canada, where his thesis concerns the development of efficient parallel numerical methods for the simulation of electrical activity in human myocardial tissue.

Raymond J. Spiteri received the Ph.D. degree in mathematics from the University of British Columbia, Vancouver, in 1997.

He is currently a Professor in the Department of Computer Science, University of Saskatchewan, Saskatoon, Canada. His research interests include numerical analysis and scientific computation, with specific interest in the development of problem-solving software environments for numerical time integration methods.