



Efficient Integration of Coupled Electrical-Chemical Systems in Multiscale Neuronal Simulations

Ekaterina Brocke^{1,2,3*}, Upinder S. Bhalla^{2†}, Mikael Djurfeldt^{4,5†},
Jeanette Hellgren Kotaleski^{1,6,7} and Michael Hanke⁸

¹ Science for Life Laboratory, Computational Science and Technology, School of Computer Science and Communication, KTH Royal Institute of Technology, Stockholm, Sweden, ² National Centre for Biological Sciences, Bangalore, India, ³ Manipal University, Manipal, India, ⁴ PDC Center for High-Performance Computing, KTH Royal Institute of Technology, Stockholm, Sweden, ⁵ International Neuroinformatics Coordinating Facility, Karolinska Institute, Stockholm, Sweden, ⁶ Science for Life Laboratory, Department of Numerical Analysis and Computer Science, Stockholm University, Stockholm, Sweden, ⁷ Department of Neuroscience, Karolinska Institute, Stockholm, Sweden, ⁸ Department of Mathematics, School of Engineering Sciences, KTH Royal Institute of Technology, Stockholm, Sweden

OPEN ACCESS

Edited by:

Florentin Wörgötter,
University of Göttingen, Germany

Reviewed by:

Gregor Kovacic,
Rensselaer Polytechnic Institute, USA
Robert H. Lee,
Emory University, USA

*Correspondence:

Ekaterina Brocke
ekaterina.brocke@scilifelab.se

[†]These authors have contributed
equally to this work.

Received: 08 March 2016

Accepted: 26 August 2016

Published: 12 September 2016

Citation:

Brocke E, Bhalla US, Djurfeldt M,
Hellgren Kotaleski J and Hanke M
(2016) Efficient Integration of Coupled
Electrical-Chemical Systems in
Multiscale Neuronal Simulations.
Front. Comput. Neurosci. 10:97.
doi: 10.3389/fncom.2016.00097

Multiscale modeling and simulations in neuroscience is gaining scientific attention due to its growing importance and unexplored capabilities. For instance, it can help to acquire better understanding of biological phenomena that have important features at multiple scales of time and space. This includes synaptic plasticity, memory formation and modulation, homeostasis. There are several ways to organize multiscale simulations depending on the scientific problem and the system to be modeled. One of the possibilities is to simulate different components of a multiscale system simultaneously and exchange data when required. The latter may become a challenging task for several reasons. First, the components of a multiscale system usually span different spatial and temporal scales, such that rigorous analysis of possible coupling solutions is required. Then, the components can be defined by different mathematical formalisms. For certain classes of problems a number of coupling mechanisms have been proposed and successfully used. However, a strict mathematical theory is missing in many cases. Recent work in the field has not so far investigated artifacts that may arise during coupled integration of different approximation methods. Moreover, in neuroscience, the coupling of widely used numerical fixed step size solvers may lead to unexpected inefficiency. In this paper we address the question of possible numerical artifacts that can arise during the integration of a coupled system. We develop an efficient strategy to couple the components comprising a multiscale test problem in neuroscience. We introduce an efficient coupling method based on the second-order backward differentiation formula (BDF2) numerical approximation. The method uses an adaptive step size integration with an error estimation proposed by Skelboe (2000). The method shows a significant advantage over conventional fixed step size solvers used in neuroscience for similar problems. We explore different coupling strategies that define the organization of computations between system components. We study the importance of an appropriate

approximation of exchanged variables during the simulation. The analysis shows a substantial impact of these aspects on the solution accuracy in the application to our multiscale neuroscientific test problem. We believe that the ideas presented in the paper may essentially contribute to the development of a robust and efficient framework for multiscale brain modeling and simulations in neuroscience.

Keywords: multiscale modeling, multiscale simulation, co-simulation, coupled system, adaptive time step integration, backward differentiation formula, parallel numerical integration, coupled integration

1. INTRODUCTION

The concept of multiscale modeling is used in many fields such as meteorology (Shukla, 2009; Kurowski et al., 2013), cardiac physiology (Hernández et al., 2011) and neuroscience (Bhalla, 2014). It refers to the style of modeling in which different models, possibly described by different physical formalisms and acting on different temporal and spatial scales, are used simultaneously in order to study important features of a complex phenomenon at multiple levels of organization (Djurfeldt and Lansner, 2007).

An idea to study how one level of neural organization influences another has become an important trend in neuroscience. For example, Bhalla (2011) explores cross-scale interactions between cellular and subcellular levels in the context of homeostasis and synaptic plasticity. This multiscale model proposes a pruning mechanism for weak synapses during cellular excitability. Another study by Mattioni and Le Novère (2013) presents an integration of electrical and biochemical processes in a model of a Medium Spiny Neuron (MSN). In particular, the influence of different input patterns on membrane excitability and the mechanism of inter-spine synaptic plasticity is discussed.

This paper focuses on the numerical aspects of different coupling strategies. The goal of the paper is to present an efficient method applicable to a wide class of problems where electrical activity and chemical signals must be solved together in the cell. This refers to almost all classes of excitable cells such as muscle fibers and nerve cells or neurons. In the neuronal context, the class of problems includes synaptic plasticity, excitability homeostasis, development and other specific functions that involve such multiscale cellular events. Here, we present a method that allows us to bridge subcellular and cellular level models described by ODEs in an accurate and efficient way.

How should the numerical solution of a multiscale system be arranged in practice? Different components of the system may need to be solved using different numerical methods. This can be the case if the components are described by different physical formalisms, for example one employs ordinary differential equations (ODEs) and the other is stochastically formulated. Even in the case when a multiscale system is formulated by a single formalism it may become advantageous to treat different components separately, both with regard to their description and with regard to computing the solution. If different components of a system are treated as independently as possible, computations can be arranged to take advantage of modularity and parallelism.

Kübler and Schiehlen (2000) propose that a complex engineering system can be decomposed into modules at three

levels of description: the physical, mathematical and behavioral levels. The components of a system can be integrated at the mathematical level and then solved using a single numerical solver. Here, we rather focus on coupling at the behavioral level—the level of signals in the computed solution. In particular, we are interested in coupling different numerical solvers in a theoretically proper and efficient way.

There are a few considerations that have to be taken into account while coupling the components at the behavioral level. First, a minimal set of signals which need to be communicated between system components has to be defined. Then, an organization between system components play a crucial role in an accurate and efficient integration of the coupled system. Mattioni and Le Novère (2013) proposed an event-driven algorithm where the exchange of the scaled variables have to be communicated each time the event happens. The algorithm showed better performance results in comparison with the time-driven algorithm where the communication of exchanged variables has to be performed at regular time intervals. However none of the recent studies give a theoretical background for the proposed coupling strategy.

One of the problems which can occur when coupling system components is numerical instability (Arnold and Günther, 2001). Numerical methods must ensure convergence of the discrete system. Even if a coupled integration is convergent, a proper choice of the step size still has to be made in order to guarantee numerical stability. It is important to keep in mind that the numerical stability of a coupled integration is not guaranteed by the stability of an independent integration of the system components (see Supplementary Instability Example). At the same time, it is crucial to perform integration in an efficient way while keeping the accuracy within desired bounds. Finally, the complexity increases when system components are described by different physical formalisms (Alfonsi et al., 2005; Rüdiger et al., 2007; Brandi et al., 2011). For example, chemical interactions can be described either in a deterministic or in a stochastic way possibly accompanied by diffusion processes.

We begin with a general introduction to the modeling at different levels within the scope of interest. In particular, in Section 2.1, we give an overview of some widely used numerical discretization methods in Neuroscience. Section 2.2 covers different numerical aspects of a coupled integration such as efficiency, order of accuracy and numerical stability. In Section 2.3, we introduce an algorithm for an adaptive control of the step size in a coupled integration. In Section 2.4, we present possible organization strategies in the system composed of multiple components. Section 2.6 describes the

multiscale test model used for our analysis: the dynamics, communication signals between the components and mathematical formulation. The details of implementation and evaluation can be found in Sections 2.7 and 2.8, respectively. In Sections 3 and 4, we present and discuss the results.

2. MATERIALS AND METHODS

2.1. Modeling at Different Levels

Here, we focus on cellular and subcellular (molecular) levels of neuronal organization. On the cellular level the object of study is typically electrical properties of a cell. The cell with its complex arborizations is usually represented by a cable split into a number of compartments. Modelers then employ compartmental modeling (Rall, 1964) to describe the neural processes where the dynamics of each compartment is defined by a system of ordinary differential equations (ODEs). Finally, by applying the Hodgkin and Huxley formalism to define the currents (Hodgkin and Huxley, 1952) a nonlinear system of ODEs has to be solved. This approach is the basis of most simulators that take neural morphology into account [e.g., NEURON (Hines, 1989), GENESIS (Wilson et al., 1990), MOOSE (Dudani et al., 2009)].

On a molecular level the interaction of biochemical signaling pathways is of particular interest. A signaling pathway is usually considered as a set of reactions between the molecules that operate on a subcellular level (Bhalla, 1998). Chemical rate theory is normally applied to describe the chemical kinetics of a signaling pathway. One of the traditional ways to model the kinetics is by viewing the system of reactions as deterministic. Then the chemical species are typically modeled as concentrations that evolve over time. These models are described by a system of nonlinear ODEs.

The complexity of the systems requires numerical computations. Simulation packages offer the user a choice between different numerical integration methods. The method choice is usually dependent on the properties of a system. For example, if the system of ODEs is *stiff* and an explicit numerical method has been chosen, the step size of the discretization is limited by stability and not by accuracy. Thus an efficient numerical integration rather requires *implicit* methods. Implicit methods often allow the simulation to be discretized with larger time steps due to good stability properties.

Fixed step size numerical methods, such as the modified Crank-Nicholson method and the classical Runge-Kutta method, are typically applied to solve systems of current study in neuroscience. The choice of the discretization time step is then made by running the simulation with different step sizes and comparing the computation cost vs. solution accuracy.

The Crank-Nicholson (CN) method with a staggered time step approach is widely used for solving branched nerve equations in neuroscience (Hines, 1984). The proposed approach allows the user to obtain a solution in an accurate and efficient way. The Crank-Nicholson method is an implicit method and can be used for stiff systems. However little is known about stability properties while working on a staggered grid (see Supplementary Numerical Methods).

For the models defined on a subcellular level, the Classical Runge-Kutta (RK4) numerical approximation method got its wide application among computational neuroscientists. It is an explicit numerical method with a bounded stability domain and therefore it is not suitable for stiff problems. A mathematical formulation of the method can be found in Supplementary Numerical Methods.

While both the CN on a staggered grid and the RK4 methods provide efficient means for simulating the models on the appropriate levels, the coupling of these methods poses additional questions.

2.2. Numerical Considerations in a Coupled Integration

2.2.1. Efficiency

Multiscale systems are usually composed of components acting on different timescales. For example, the timescale of a single spike is of the order of a few milliseconds. However, simulations normally run for many seconds in order to observe the effects at a biochemical level. The gap between timescales demands an efficient integration strategy. We suspect that adaptive integrators can be more efficient or require less time for a given degree of accuracy. Both the CN on a staggered grid and the RK4 methods are normally applied on a fixed step size grid. Neither provide an error estimation that could be used together with a step size control mechanism.

2.2.2. Order of Accuracy

Little is known about the error propagation for the CN on a staggered grid with RK4 coupled integration. The order of the numerical method quantifies the global error behavior with respect to the step size. We know that the CN on a staggered grid method is second-order accurate and the RK4 method is of the order four while the order of the coupled integration still has to be studied.

2.2.3. Numerical Stability

Numerical properties, such as numerical stability, of a numerical algorithm applied to each system component independently may not hold in a coupled integration. For example, the Backward Euler method applied to a stable system may lead to an amplified oscillating behavior of the system when solved in a coupled manner as shown in Supplementary Instability Example. This may happen due to the different component properties, such as stiffness, that may arise in a coupled system simulation. Our multiscale system is often stiff due to the rapid changes in the electrical component. Thus, to be on the safe side we aim to avoid using methods not suitable for stiff problems.

2.3. Numerical Algorithm

2.3.1. Backward Differentiation Formula

Considering the thoughts above, we are interested in another family of implicit methods, Backward Differentiation Formula (BDF). In particular, the second-order BDF (BDF2) method has gained a large recognition in its application to stiff differential equations and Differential Algebraic Equations (DAEs). Moreover, Skelboe (2000) provides stability analysis and

error estimates within a coupled numerical integration strategy. In Skelboe (2000) refers to the method as to the decoupled BDF2 formula. According to the algorithm a given system has to be partitioned into loosely coupled subsystems first and then the decoupled formula is applied to integrate the system. Here, the subsystems to be coupled are given beforehand. A mathematical formulation of the method is given in Supplementary Numerical Methods.

2.3.2. Adaptive Step Size Controller

The aim of the adaptive step size controller is to reduce the computational cost of the simulation while keeping the local error within acceptable bounds. The reduced computational cost is mainly achieved by a reduced number of time steps at which a solution is approximated. In particular, the step size controller aims at guaranteeing Equation (1) for an as large as possible step size h , where $|\epsilon_j|$ is the local discretization error and the parameter tolerance TOL is a bound on the local error.

$$|\epsilon_j| \leq TOL \quad (1)$$

We implement an adaptive control of the step size shown in Algorithm 1. It calculates a local discretization error $|\epsilon_j|$ of the taken step h_j (line #8). If the calculated quality is good enough (line #9), an optimal step size is calculated and then used as a predictor for the next step size h_{j+1} (line #13). Otherwise, the current step size h_j is recalculated (line #16) and the step has to be done once again.

Since the quantity of the local discretization error $|\epsilon_j|$ cannot be determined *exactly*, the notion of some *computable estimate* $|\epsilon_j| \approx |\hat{\epsilon}_j|$ was introduced by Deuffhard and Bornemann (2002).

In Algorithm 1 we apply two controllers depending on the computed local discretization error value to predict an optimal time step, a so-called H211b controller proposed by Söderlind and Wang (2006) and a P-controller used by Deuffhard and Bornemann (2002, pp. 197ff). The optimal time step is also bounded by qh_j ($q > 1$) and h_{max} terms for the situation when the error estimation value becomes or is close to zero (line #13 and line #16).

In order to implement Algorithm 1 an error estimation mechanism (line #8) should be provided.

2.3.3. Estimate of the Local Discretization Error

The error estimation is method dependent. Below we describe the error estimation for the BDF2 method based on a predictor-corrector algorithm. The difference of the discrete evolutions given by Equation (2) represents an estimate of the local error $|\epsilon|$:

$$|\epsilon_j| = \left\| \Psi^{t+h,t} x - \hat{\Psi}^{t+h,t} x \right\|, \quad (2)$$

where the prediction step taken by the discrete evolution $\hat{\Psi}^{t+h,t}$ calculates a rough approximation of the solution and the corrector step taken by $\Psi^{t+h,t}$ refines the initial approximation.

Often, the tolerance TOL is set as a combination of a relative tolerance, $relTOL$, and an absolute tolerance, $absTOL$. Then,

Algorithm 1 Step size controller

```

1:  $h_0 = \bar{h}_0$  % Choose initial step size prediction  $\bar{h}_0$ 
2:  $j = 0$  % Initiate the iteration index
3:  $\Delta_t = \{t_0\}$  % Initiate the time set
4:  $x_\Delta(t_0) = x_0$  % Initiate the solution set
5: while  $t_j < T$  do % Within the simulation time  $T$  do:
6:    $t = t_j + h_j$ 
7:    $x = \Psi^{t,t_j} x_\Delta(t_j)$  % Advance the solution from  $t_j$  to  $t$ 
8:   compute the error estimate  $|\epsilon_j|$ 
9:   if  $|\epsilon_j| \leq TOL$  then % Step is accepted
10:     $t_{j+1} = t$ 
11:     $\Delta_t = \Delta_t \cup \{t_{j+1}\}$  % Update the time set
12:     $x_\Delta(t_{j+1}) = x$  % Update the solution set
13:     $h_{j+1} = \min(qh_j, h_{max},$ 
       $H211b(h_j, |\hat{\epsilon}_j|),$ 
       $(T - t_{j+1}))$ 
      % Calculate an optimal time step  $h_{j+1}$ 
      % using Söderlind H211b controller
      (Söderlind and Wang, 2006)
14:     $j = j + 1$ 
15:   else % Step is rejected
16:     $h_j = \min(qh_j, h_{max},$ 
       $\sqrt[p+1]{\frac{\rho TOL}{|\epsilon_j|}} h_j)$ 
      % Adjust the current time step  $h_j$ 
      % using a P-controller
      (Deuffhard and Bornemann, 2002, pp. 197ff)
17:   end if
18: end while

```

considering Equations (1) and (2) we can write:

$$\left\| \Psi^{t+h,t} x - \hat{\Psi}^{t+h,t} x \right\| \leq relTOL \cdot \left\| \Psi^{t+h,t} x \right\| + absTOL \quad (3)$$

From a practical point of view, it is more appropriate to use a similar requirement component-wise. By rearranging Equation (3) we obtain the error control quantity $|\epsilon|$ with the bound equal to one:

$$|\epsilon| = \max_i \frac{\left| \Psi^{t+h,t} x_i - \hat{\Psi}^{t+h,t} x_i \right|}{relTOL \cdot \left| \Psi^{t+h,t} x_i \right| + absTOL_i} \leq 1 \quad (4)$$

where i is the index of the integrated variable in the solution vector.

The best efficiency is usually achieved if both the predictor and the corrector are of the same order (Sjö, 1999). Furthermore polynomial interpolation formulas are preferred as predictors in connection with stiff problems (Skelboe, 2000). The prediction step in Equation (4) is taken by the discrete evolution of the second-order polynomial described in Section 2.5, and the corrector step by the BDF2 method.

2.4. Organization of System Components in a Coupled Integration

In the coupled integration the components of a system are solved independently on time windows $[T_n, T_{n+1}]$. Then information is

exchanged at communication points $0 = T_0 < \dots < T_n < T_{n+1} < \dots$. Different aspects of integration are considered in the literature: the use of multiple discretization step sizes (*multirate* methods), the coupling of different numerical methods (*multi-method* integration) and different types of organization of computations between the components. Here, we study two organizations: *Jacobi* and *Gauss-Seidel*. The choice may have a crucial impact on both numerical stability and accuracy.

We introduce the notion of *macro time step* and *micro time step*. The macro time step determines the communication points: how long the components can run independently from each other without losing accuracy. The micro time step determines the discretization points of each component between two communication points. The latter is usually determined by accuracy, stability and thus the numerical method used.

For the purpose of explaining Jacobi and Gauss-Seidel organizations we introduce an abstract system composed of two components: component 1 and component 2. Then we can define the system using a continuous representation as:

$$\begin{aligned} \frac{d}{dt}x_1 &= f_1(t, x_1, x_2) \\ \frac{d}{dt}x_2 &= f_2(t, x_1, x_2), \end{aligned} \tag{5}$$

where x_1, x_2 are solution vectors of the respective component. Note that although the organization is discussed by considering only two components, the basic principles carry over the general case of n components.

2.4.1. Jacobi Organization

Jacobi organization in the system with two components leads to the interaction shown in **Figure 1**. In order to make a step from time T_n to T_{n+1} each component gets variables exchanged at time T_n (white triangle arrows on **Figure 1**). Then, the components are integrated to the time point T_{n+1} .

In Jacobi organization System (5) can be rewritten in the form:

$$\begin{aligned} \frac{d}{dt}x_1 &= f_1(t, x_1, \tilde{x}_{2,n+1}) \\ \frac{d}{dt}x_2 &= f_2(t, \tilde{x}_{1,n+1}, x_2), \end{aligned} \tag{6}$$

where $\tilde{x}_{1,n+1}$ and $\tilde{x}_{2,n+1}$ are approximations of the exchanged variables x_1 and x_2 at time T_{n+1} , respectively. This organization works well in parallel computations since no component needs to wait for the other.

2.4.2. Gauss-Seidel Organization

The Gauss-Seidel organization updates each component in a sequential order (**Figure 2**). Let the component 1 be the leading component in System (5). Then, after the solution has been communicated at time T_n (white triangle arrow on **Figure 2**), the component 1 proceeds until T_{n+1} . Then the calculated solution of the component 1 at time T_{n+1} can be used by the component 2. This organization has been used by Mattioni and Le Novère (2013).

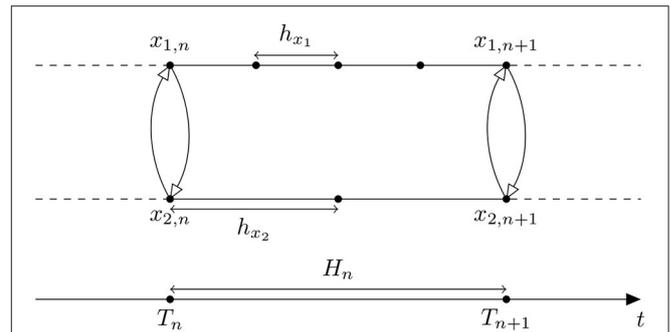


FIGURE 1 | Discretization in time of System (5) using Jacobi organization. White triangle arrows correspond to the variables exchanged between component 1 and component 2. H_n is a macro time step of System (5), h_{x_1}, h_{x_2} are micro time steps of the component 1 and 2 respectively.

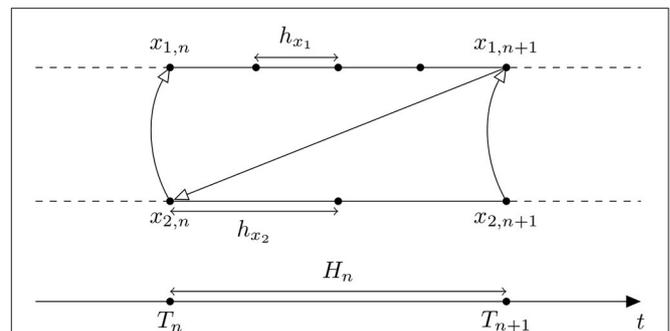


FIGURE 2 | Discretization in time of System (5) using Gauss-Seidel organization. White triangle arrows correspond to the variables exchanged between the component 1 and the component 2. H_n is a macro time step in System (5), h_{x_1}, h_{x_2} are micro time steps of the component 1 and 2, respectively.

The Gauss-Seidel organization allows to eliminate the error introduced by one solution approximation in System (7).

$$\begin{aligned} \frac{d}{dt}x_1 &= f_1(t, x_1, \tilde{x}_{2,n+1}) \\ \frac{d}{dt}x_2 &= f_2(t, x_1, x_2), \end{aligned} \tag{7}$$

where $\tilde{x}_{2,n+1}$ is an approximated solution of the component 2 as in Equation (6). However while Jacobi organization preserves parallelity of the integration, the Gauss-Seidel organization imposes a sequence of computation which reduces parallelity (Skelboe, 1992).

2.5. Approximation of Exchanged Variables

We aim to simulate two components with reciprocal data dependencies, that is the solution of each component depends on the solution of the other component at each integration time step. It can happen that the information is not available at a certain time point. For instance, the implicit methods require exchanged variable values and state variables from the current

time point T_{n+1} as shown in System (6) and in System (7). Then an approximation of exchanged variables can be considered.

Here we will compare two approximation strategies, so called *Mode 1* and *Mode 3* introduced in Skelboe (2000). Mode 1 implies a constant extrapolation $\tilde{x}_{n+1} \equiv x_n$, so that the solution at the previous time step n is used when required. A second-order polynomial is applied in Mode 3. It uses the data from the previous time steps to approximate the value at time T_{n+1} Equation (8).

$$\begin{aligned}\tilde{x}_{n+1} &= x_{n+1}^{p2} = \bar{\alpha}_1 x_n + \bar{\alpha}_2 x_{n-1} + \bar{\alpha}_3 x_{n-2} \\ \bar{\alpha}_1 &= 1 - \bar{\alpha}_2 - \bar{\alpha}_3, \bar{\alpha}_2 = \frac{\gamma_{n+1}(\gamma_{n+1} + \delta_{n+1})}{1 - \delta_{n+1}}, \\ \bar{\alpha}_3 &= \frac{\gamma_{n+1}(\gamma_{n+1} + 1)}{\delta_{n+1}(\delta_{n+1} - 1)},\end{aligned}\quad (8)$$

where $\gamma_{n+1} = h_{n+1}/h_n$ and $\delta_{n+1} = 1 + h_{n-1}/h_n$. In general, Equation (8) can be easily extended to a continuous extrapolation. In that case, the parameters $\bar{\alpha}_1$, $\bar{\alpha}_2$, $\bar{\alpha}_3$ are functions of time. We use Equation (8) as a predictor in the estimate of the local discretization error and as an approximator of the exchanged variables in Mode 3.

2.6. Test Case

In computational neuroscience both the electrical and the chemical signaling play a crucial role in the studies of learning and memory mechanisms. Synaptic input integration, signal propagation along the dendrites and action potential generation can often be modeled as electrical processes. These processes act on a timescale of about 0.001 s and a spatial scale from a few 100 μm to a few millimeters. The chemical processes typically span the timescales from a few seconds to months and years and can act within 1 μm . In our multiscale test system we span two levels of neural organization: cellular and subcellular levels. We model the electrical dynamics of a single neuron and biochemical processes in its *spine* (Figure 3).

Models of bistable chemical switches are commonly explored in the studies of long-term memory formation, in particular LTP and LTD. We choose the mitogen-activated protein kinase (MAPK) signaling model used in the study of homeostatic regulation of excitability at the scale of a single synapse by Bhalla (2011). The model exhibits bistability. It changes from the inactive to the active stable state by means of receptor phosphorylation caused by a calcium signal. To simulate the calcium influx we model a compartmentalized cell of a regular-spiking neuron (Pospischil et al., 2008). A current of $0.09 \cdot 10^{-9}$ A is applied to the *soma* of the cell for 5 s. The change in the membrane potential activates voltage dependent sodium and potassium channels and a spike train propagates through the axial resistance to the spine. Spine depolarization activates voltage dependent calcium channels and a calcium current flows into the spine. Calcium influx triggers multiple signaling cascades on a sub-cellular level. *P-MAPK* becomes active, phosphorylates potassium channels K_A and leads to their non-conductivity. Then the biochemical model settles at its second steady state. The parameters of both models are given in Supplementary Model Details.

2.6.1. Communication Signals

We use Ca^{2+} influx as a key signal in our multiscale model to activate the MAPK cascade. The Ca^{2+} current in the electrical model (I_{Ca}) is transformed to the calcium injection rate to the biochemical model (k_{inj}) as shown in Equation (9).

$$k_{inj} = \frac{N_e}{2 \cdot N_A \cdot vol} \cdot I_{Ca} \quad \left[\frac{M}{s} \right], \quad (9)$$

where N_e is the number of electrons in one Coulomb which roughly equals $6.242 \cdot 10^{18}$, N_A is Avogadro's constant and vol is the volume of the spine compartment.

In turn, the biochemical model provides calcium concentration $[Ca]$ and the fraction of active (non-phosphorylated) potassium channels in the spine $\frac{[K_A]}{[K_{base}]}$. The fraction is used in the conductance evolution of the A-type potassium current in the electrical model Equation (10).

$$g_{K_A} = \bar{g}_{K_A} \frac{[K_A]}{[K_{base}]} \quad [S], \quad (10)$$

where \bar{g}_{K_A} is the maximum potassium conductance. Calcium concentration $[Ca]$ is used in the Nernst potential recalculations for the calcium ion in the spine (see Supplementary Model Details).

2.6.2. Mathematical Formulation

The compartmental modeling with the HH formalism in the electrical component defines 17 subsequent electrical circuits or 24 ODEs (see Supplementary Model Details). Chemical reactions in the spine are defined by reaction-rate equations constituting a non-linear system of 18 ODEs (see Supplementary Model Details). Considering the communication signals System (5) can be reformulated in the following form:

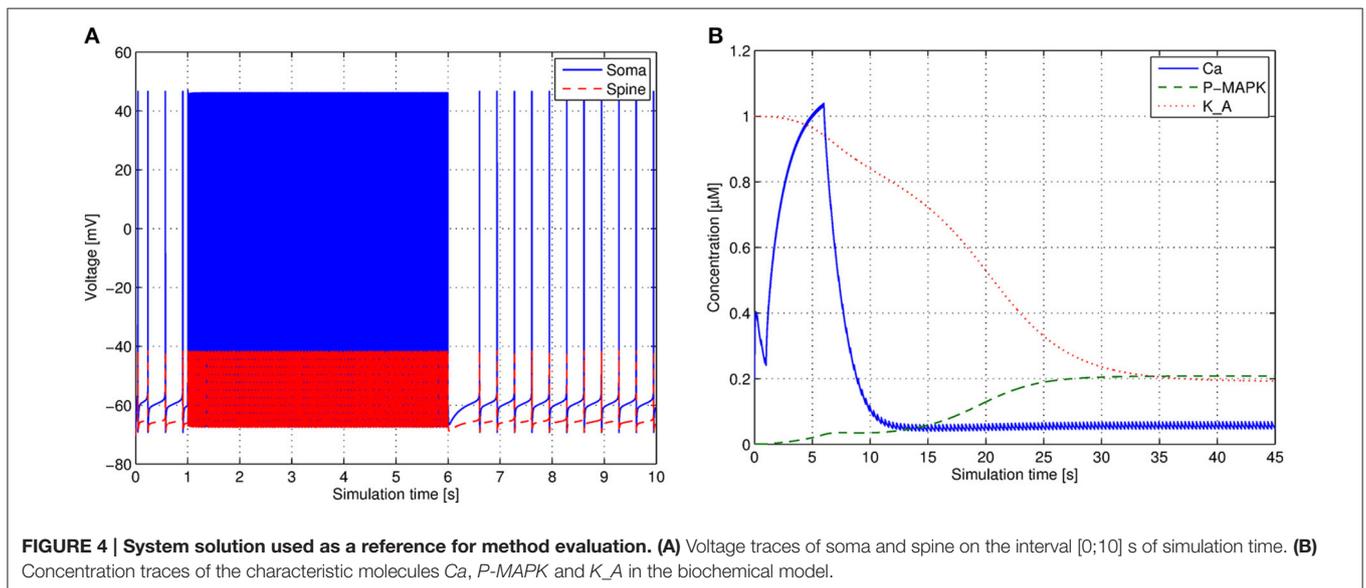
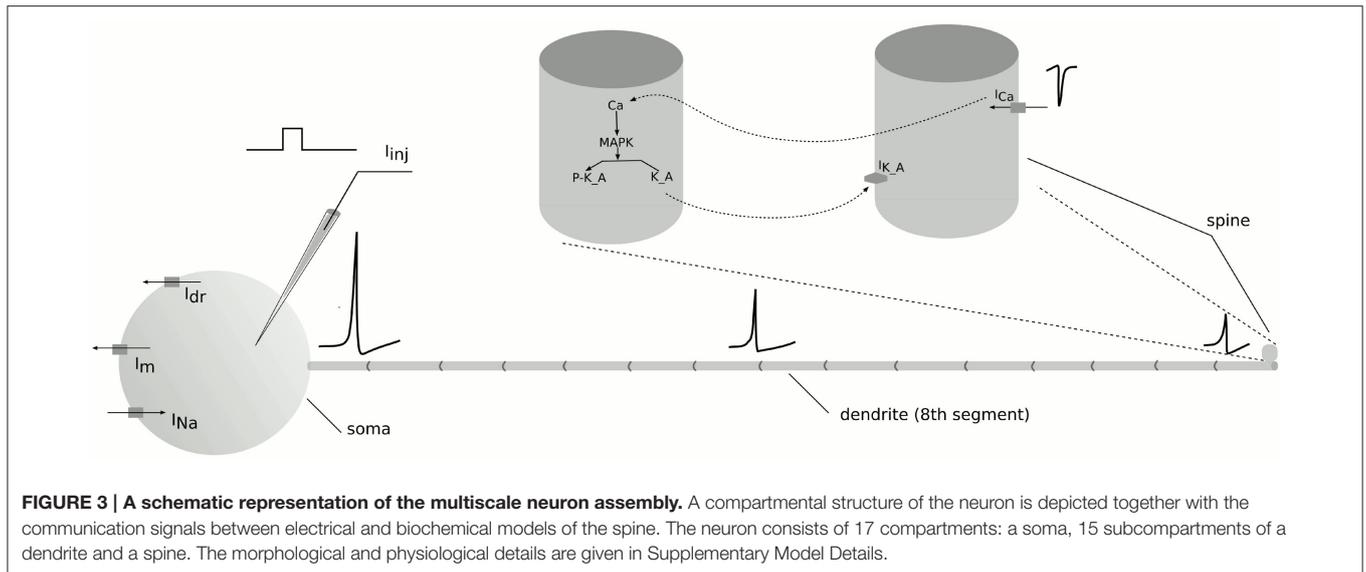
$$\begin{aligned}\frac{d}{dt} x_{chem} &= f_{chem}(t, x_{chem}, g_1(x_{elec}, x_{chem})) \\ \frac{d}{dt} x_{elec} &= f_{elec}(t, g_2(x_{chem}), x_{elec}),\end{aligned}\quad (11)$$

where g_1 and g_2 are the output functions from the electrical and the biochemical component respectively.

$$\begin{aligned}g_1(x_{elec}, x_{chem}) &= C_1(x_{elec,i} + C_2 \ln(x_{chem,j}) - C_3)x_{elec,k}^2 \\ g_2(x_{chem}) &= \begin{cases} x_{chem,m} \\ C_4 x_{chem,n} \end{cases},\end{aligned}\quad (12)$$

where $C_1..C_4$ are constants; the indices $i..m$ correspond to positions of variables in the solution vector x at time t :

- i - potential in the spine [V];
- j, m - calcium concentration in the spine [M];
- k - probability for an s gate being opened (calcium channel activation);
- l - probability for an r gate being opened (calcium channel inactivation);
- n - concentration of active (non-phosphorylated) potassium channels in the spine [M].



2.7. Implementation and Simulation

The electrical and biochemical models, numerical discretization methods, Crank-Nicholson on a staggered grid, RK4 and BDF2, along with the adaptive step size controller was implemented in MATLAB[®]. The Crank-Nicholson on a staggered grid and the RK4 methods were used to solve the electrical and the biochemical component respectively. The adaptive step size controller was used only when both the electrical and the biochemical models were discretized with the BDF2 method.

Often a reference solution of the system is required for the numerical method evaluation. To obtain the reference solution we solve our system using *ode15s* MATLAB[®] function. *ode15s* is a solver designed for stiff problems. It is a quasi-constant step size implementation of the backward differentiation methods (Shampine and Reichelt, 1997). We set *RelTol* and

AbsTol parameters to 10^{-9} and 10^{-12} respectively to achieve high accuracy in the reference solution. The **Figures 4A,B** depict the traces of a few most representative variables from the electrical and the biochemical components in the system respectively.

In our simulations in the proposed methods we used macro time step equal micro time step, that is $H_n = h_{x_1} = h_{x_2}$. In general, the macro time step does not have to be equal the micro time step. For example, one can imagine to reduce the number of communication points when appropriate in order to optimize the data flow in parallel simulations. Another example is the multirate methods where each component can be solved with its own discretization time step, that is $h_{x_i} \neq h_{x_j}$. In this case, the macro time step is different from the micro time step at least for one of the components.

Two input parameters, $relTOL$ and $absTOL$, are required by the error estimation step in the adaptive algorithm. The latter one, the absolute error tolerance $absTOL$, determines the accuracy when the solution approaches zero. In order to reduce the number of parameters we replaced $absTOL$ in Equation (4) with the relative tolerance multiplied with a value representing a typical size of the corresponding solution component, $absTOL_i = relTOL \cdot |Y_{typical, i}|$. We assigned $Y_{typical}$ to the reference solution vector chosen at an arbitrary point of time during the stimulation.

2.8. Evaluation

We apply visual inspection to estimate the global error of the method by plotting the order-relative slopes, first- and second-order declines, in each figure in Section 3 (dashed lines). Visual inspection can also be used to understand whether an obtained solution deviates from the reference one. However in order to compare the accuracy of the methods a more rigorous technique is required.

A typical way to evaluate the accuracy of a numerical approximation method is to calculate the relative error of the solution at a characteristic point (often taken to be the final) of the simulation. In our system, the calcium communication signal has a crucial impact on the biochemical component. Thus we took a solution of the calcium concentration to calculate the error. To verify our observations we applied the same evaluation technique to the voltage solution in the spine.

We propose a component-wise relative error represented by Equation (13).

$$\epsilon = \frac{1}{N} \sum_i^N \|\vec{r}_i\| \cdot 100 \text{ [%]} \quad (13)$$

where $\vec{r}_i = (\check{p}_i - p_i) \oslash \check{p}_i$, $\check{p}_i = (t_{\check{p}_i}, v_{\check{p}_i})$ and $p_i = (t_{p_i}, v_{p_i})$ are characteristic points on the reference and approximate solutions respectively and \oslash stands for an element-wise division of the vectors. Thus two components, the time t and the magnitude v , are considered in our formula. We motivate this choice with the thought in mind that both the strength and the time of the event are important in our simulations. In our solutions we defined maxima on an arbitrary chosen interval of the simulation as our characteristic points. We located two maxima values ($N = 2$) on the interval between 1 and 2 s of the simulation time to calculate relative error ϵ Equation (13).

To look at the efficiency of the coupling method we plotted the error vs. the total number of function calls of the right hand side of the system of ODEs during the simulation time $T = 2$ s. In the BDF2 implementation the total number of function calls includes the function calls used to compute the Jacobian matrix. For the RK4 method the total number of function calls per step equals four. In the CN on staggered grid implementation we estimated the work as one function call plus one Jacobian evaluation. The latter is comparable to one function call per step.

3. RESULTS

3.1. The BDF2-BDF2 Coupling Outperforms RK4-CN on a Fixed Step Size Grid

We have a stiff electrical component that is usually solved with the CN method on a staggered grid in neuroscience. We applied the RK4 method to the biochemical system (Bhalla, 2011). Both methods are fixed step size methods. The BDF2 method can also be used on a fixed step size grid however this can be inefficient due to the multiple iterations per step required in implicit methods.

In **Figure 5** we compare the accuracy of two different coupling methods, the BDF2-BDF2 coupling method and the CN on staggered grid with RK4 coupling on a fixed step size grid.

We observe that the accuracy of the decoupled BDF2 formula is preserved in the solutions. However an expected second-order accuracy of the calcium solution obtained with the RK4-CN coupling is lost (**Figure 5A**). This can happen due to insufficient accuracy of the exchanged variables. In the modified CN method calcium current is calculated at a half time step and then communicated at a whole time step. Thus the accuracy of the communicated variable is only of the first order. This can influence the order of the coupling method. We performed simulations using different order extrapolating techniques of the exchanged variables. However, we did not notice a sufficient order difference with respect to the calcium solution.

There is no visible accuracy loss in the voltage traces as shown in **Figure 5B**. We expect the order of a coupling method to be dependent on the strength of the coupling and thus vary for different systems. In our multiscale test problem we have a strong coupling influence of the electrical component on the biochemical and a weak coupling vice versa.

Overall the BDF2-BDF2 coupling shows more accurate results than the conventional methods on a given range of fixed step size grids. These promising outcomes indicate a further direction of the research.

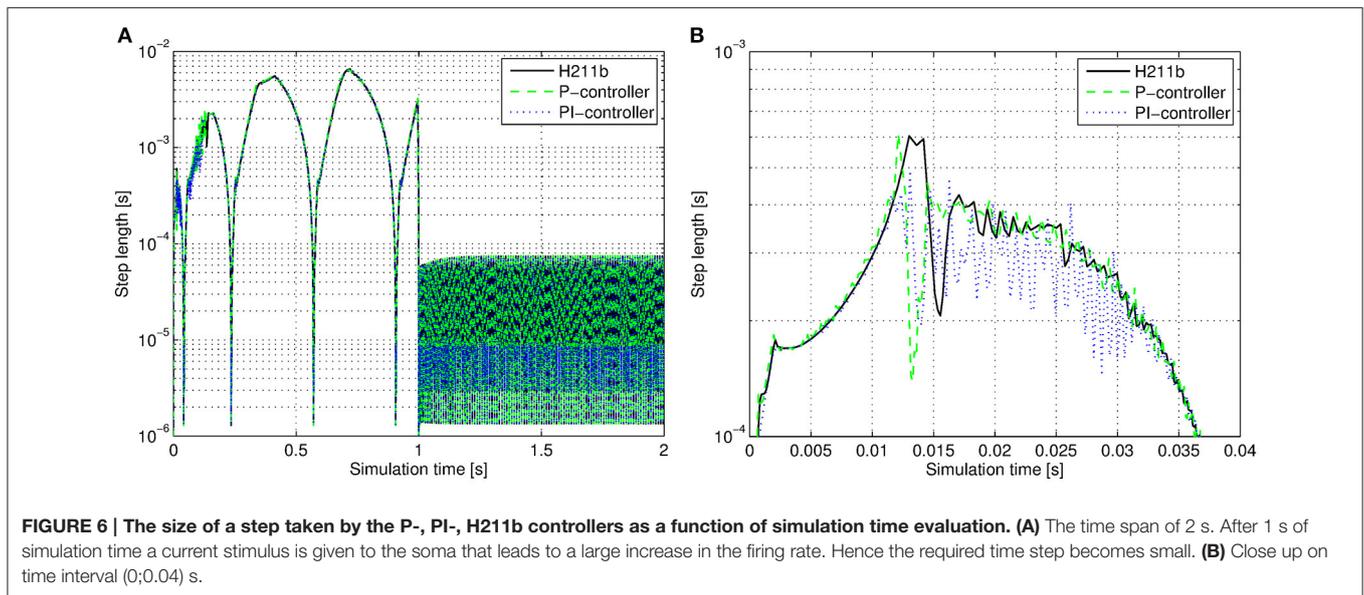
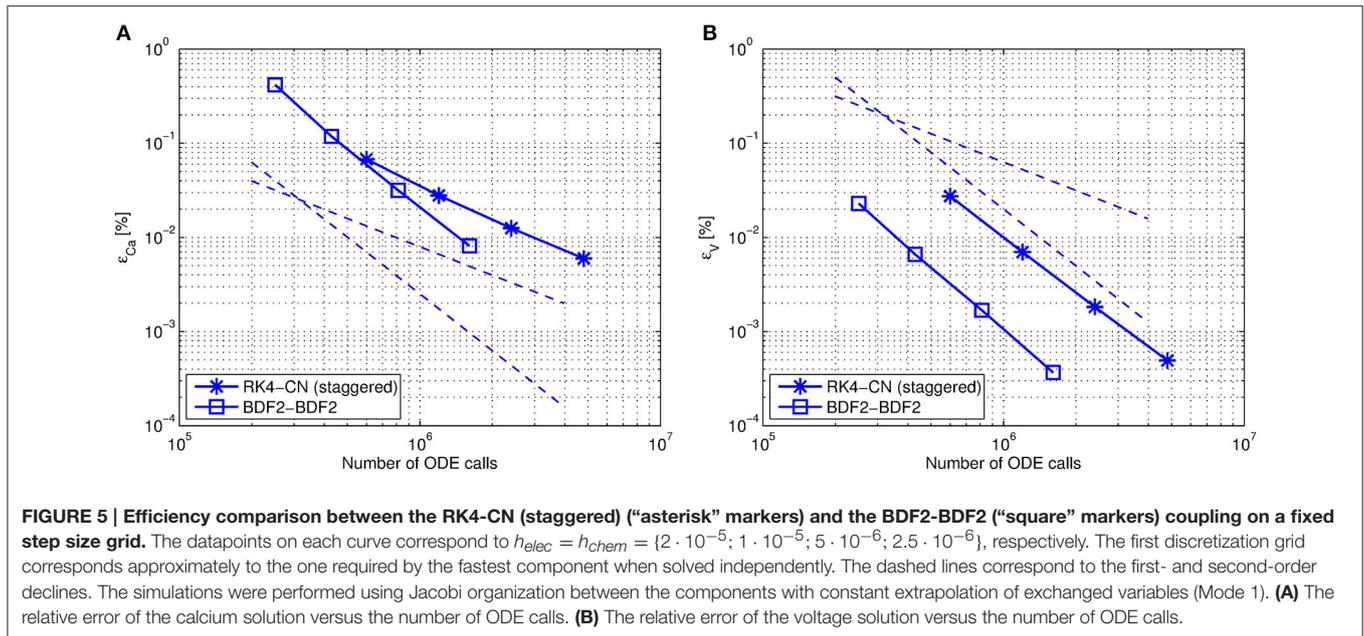
3.2. The H211b Controller Leads to a Smoother Distribution of Step Sizes

An optimal behavior of the step size controller is when the step sizes that have to be taken do not have an extensive variation. Otherwise it increases the number of times the step size controller has to redo a step. A smooth distribution of step sizes leads to good performance of the integrator.

In **Figure 6** we compare the P-, PI- and H211b controllers. The H211b controller shows a smoother step size variation at the beginning of the simulation. Thus we chose the H211b controller for our further observations.

3.3. An Appropriate Approximation of Exchanged Variables

In these simulations we compare the constant extrapolation (Mode 1) with the second-order polynomial extrapolation of exchanged variables (Mode 3) described in Section 2.5. We used Jacobi organization of the components and an



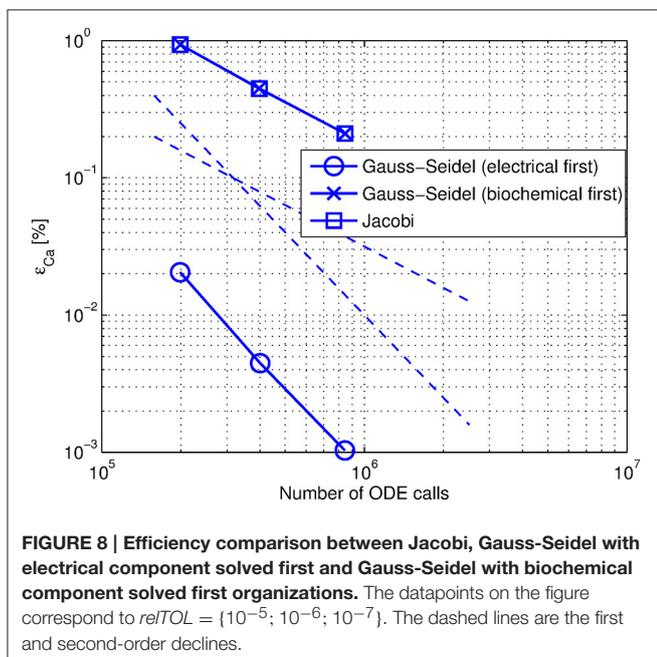
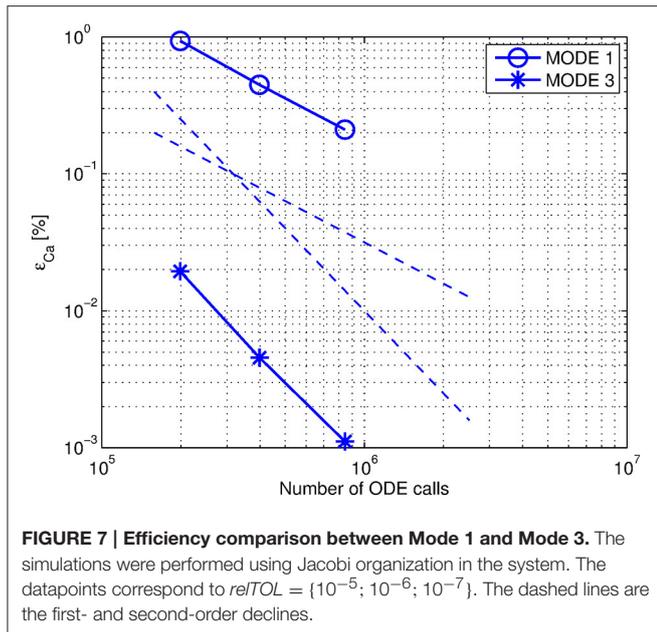
adaptive step size controller with the BDF2 approximation method in these simulations. The relative errors both of calcium concentration ϵ_{Ca} and of voltage in the spine ϵ_V were calculated. An expected asymptotic behavior is observed in Mode 3 and only a first-order of coupling in Mode 1 (Figure 7). Accuracy measurements of the voltage solution do not show significant difference between the modes (not shown).

A significant advantage of the second-order polynomial extrapolation (Mode 3) over the constant extrapolation (Mode 1) with respect to solution accuracy demonstrates the importance of an appropriate choice between different approximation strategies of exchanged variables.

In general, approximated values can be of 0th order (Mode 1) if the coupling between the components is weak or $x_i(t)$ is slowly varying, otherwise the results will be inaccurate (Sand and Skelboe, 1992). Our results are perfectly in line with this observation since our test case has a strong coupling between the electrical and the biochemical component.

3.4. Organization of System Components

We investigate whether different organizations between the components in our system have an impact on the solution accuracy. Having a strong influence of the electrical component on the biochemical during the stimulation interval we predicted



that by letting the electrical component lead the integration we could possibly avoid an approximation error of the exchanged variables and improve overall performance. The results we observe are consistent with our expectations (**Figure 8**). Gauss-Seidel organization with the electrical component solved first in Mode 1 lead to more accurate results.

We also compared different organizations in Mode 3. We did not notice any superiority of Gauss-Seidel organization with the electrical component solved first.

We conclude that Jacobi organization with the second-order polynomial approximation of the exchanged variables (Mode 3)

can have similar accuracy as Gauss-Seidel organization with an appropriate ordering of system components in Mode 1.

4. DISCUSSION

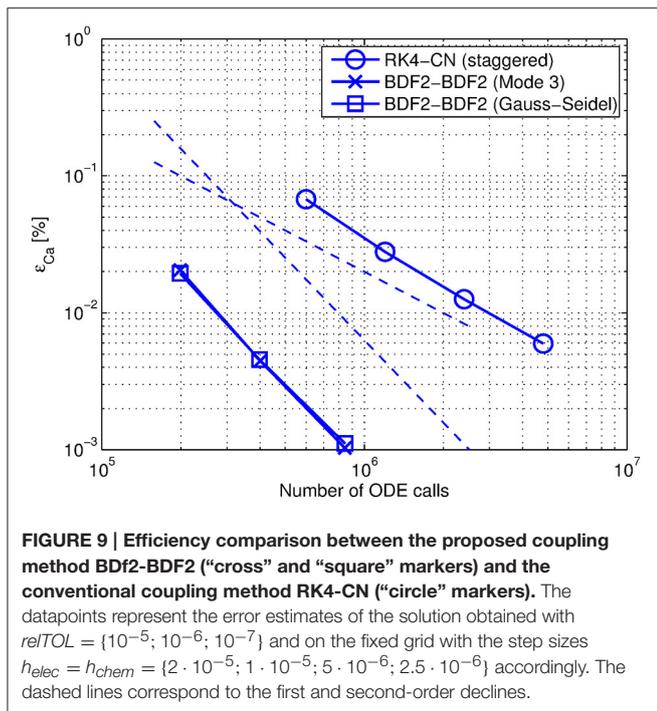
Several strategies have been proposed for coupling multiple components of a multiscale system on a behavioral level recently (Bhalla, 2011; Mattioni and Le Novère, 2013). One of the questions that has not been addressed in the research literature is mathematical justification of the coupling strategy. In this paper we discussed the problems of inefficiency and possible numerical instability that may arise while coupling multiple components comprising a multiscale system.

We introduced an implicit approximation method, two-step Backward Differentiation formula (BDF2) as a possible alternative to the conventional discretization methods used in neuroscience. This numerical method appealed to our interest for several reasons. First, it has been previously introduced as a decoupled implicit BDF2 formula by Skelboe (2000). Its stability properties and error propagation estimates within a coupled integration have been discussed. Second, the proposed error estimation allowed us to use the adaptive time-stepping algorithm described in Section 2.3.

In this paper we investigated the influence of such factors as an approximation of exchanged variables and different organizations of system components. We compared constant (Mode 1) and second-order polynomial (Mode 3) extrapolation of exchanged variables in our simulations. We also introduced two types of organization between the components, Jacobi and Gauss-Seidel. Our results show that an appropriate approximation of exchanged variables and organization of system components plays a significant role in efficient integration of the coupled system. Moreover, an application of the proposed integration method with the Jacobi-type organization of system components is well suited for parallel computations.

The power and applicability of the method was demonstrated solving a multiscale test case that was designed as a prototype of models used in the scope of interest. In **Figure 9** we combine the efficiency measurements of the solutions obtained with the adaptive step size BDF2-BDF2 coupling method and those obtained with the fixed step size coupling RK4 - CN (staggered) method. The BDF2-BDF2 coupling in Mode 3 with Jacobi organization (the curve with “square” markers) is comparable to the BDF2-BDF2 coupling in Mode 1 with Gauss-Seidel organization where the electrical component was solved first (the curve with “asterisk” markers). The proposed integration method allowed us to solve the system in a more efficient way. An application of the method and conclusions can be generalized for similar problems formulated by systems of ODEs and DAEs.

Multiscale problems usually span multiple time scales. The step sizes required for numerical stability and desired accuracy normally differ for different system components when solved separately. In this paper, we used macro time-stepping equal micro time-stepping for each component in order to address the questions of stability and accuracy of a coupled integration. Intuitively one would imagine to use small steps for the fastest



changing components and larger steps in slow components in the hope of reducing the amount of computational and communication work, a so-called *multirate method*. To the knowledge of the authors an application of the multirate integration to the physical systems is a non trivial task with a limited choice of available methods. Skelboe (2000) mentions an applicability of the decoupled integration formula in a multirate mode with a waveform relaxation method. In Günther and Rentrop (1993) tested the multirate Rosenbrock-Wanner

REFERENCES

- Alfonsi, A., Cancès, E., Turinici, G., Di Ventura, B., and Huisinga, W. (2005). “Adaptive simulation of hybrid stochastic and deterministic models for biochemical systems,” in *ESAIM: Proceedings*, Vol. 14 (Les Ulis: EDP Sciences), 1–13.
- Arnold, M., and Günther, M. (2001). Preconditioned dynamic iteration for coupled differential-algebraic systems. *BIT Numer. Math.* 41, 1–25. doi: 10.1023/A:1021909032551
- Bhalla, U. S. (1998). “The network within: signaling pathways,” in *The Book of GENESIS* (New York, NY: Springer), 169–191. doi: 10.1007/978-1-4612-1634-6_10
- Bhalla, U. S. (2011). Multiscale interactions between chemical and electric signaling in ltp induction, LTP reversal and dendritic excitability. *Neural Netw.* 24, 943–949. doi: 10.1016/j.neunet.2011.05.001
- Bhalla, U. S. (2014). Multiscale modeling and synaptic plasticity. *Prog. Mol. Biol. Transl. Sci.* 123, 351–386. doi: 10.1016/B978-0-12-397897-4.00012-7
- Brandi, M., Brocke, E., Talukdar, H. A., Hanke, M., Bhalla, U. S., Kotaleski, J. H., et al. (2011). Connecting moose and neurond through music: towards a communication framework for multi-scale modeling. *BMC Neurosci.* 12(Suppl. 1):P77. doi: 10.1186/1471-2202-12-S1-P77
- Deuffhard, P., and Bornemann, F. (2002). *Scientific Computing with Ordinary Differential Equations*, Vol. 42. New York, NY: Springer.

schemes on a highly integrated electric circuits. The method showed a potential for a computational speedup. Therefore, an extension of the proposed method to allow multirate integration can be further considered.

AUTHOR CONTRIBUTIONS

All authors listed, have made substantial, direct and intellectual contribution to the work, and approved it for publication. MH, UB, EB, MD, and JH conceived, designed the experiments. EB and MH performed the experiments. MH, EB, UB, MD, and JH analyzed the results. EB wrote the paper.

FUNDING

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no 604102 (HBP), the Swedish Research Council (Vetenskapsrådet), NIAAA (grant 2R01AA016022), Swedish e-Science Research Center, and an Erasmus Mundus Joint Doctoral program EuroSPIN.

ACKNOWLEDGMENTS

We would like to thank Jerker Nilsson who implemented a first version of the algorithm and carried out initial numerical experiments.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <http://journal.frontiersin.org/article/10.3389/fncom.2016.00097>

- Djurfeldt, M., and Lansner, A. (2007). Workshop report: 1st incf workshop on large-scale modeling of the nervous system. *Nat. Precedings*. doi: 10.1038/npre.2007.262.1
- Dudani, N., Ray, S., George, S., and Bhalla, U. S. (2009). Multiscale modeling and interoperability in moose. *BMC Neurosci.* 10(Suppl. 1):P54. doi: 10.1186/1471-2202-10-S1-P54
- Günther, M., and Rentrop, P. (1993). Multirate row methods and latency of electric circuits. *Appl. Numer. Math.* 13, 83–102. doi: 10.1016/0168-9274(93)90133-C
- Hernández, A. I., Le Rolle, V., Ojeda, D., Baconnier, P., Fontecave-Jallon, J., Guillaud, F., et al. (2011). Integration of detailed modules in a core model of body fluid homeostasis and blood pressure regulation. *Prog. Biophys. Mol. Biol.* 107, 169–182. doi: 10.1016/j.pbiomolbio.2011.06.008
- Hines, M. (1984). Efficient computation of branched nerve equations. *Int. J. Biomed. Comput.* 15, 69–76. doi: 10.1016/0020-7101(84)90008-4
- Hines, M. (1989). A program for simulation of nerve equations with branching geometries. *Int. J. Biomed. Comput.* 24, 55–68. doi: 10.1016/0020-7101(89)90007-X
- Hodgkin, A. L., and Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.* 117:500.
- Kübler, R., and Schiehlen, W. (2000). Two methods of simulator coupling. *Math. Comput. Modell. Dyn. Syst.* 6, 93–113. doi: 10.1076/1387-3954(200006)6:2;1-M:FT093

- Kurowski, M. J., Grabowski, W. W., and Smolarkiewicz, P. K. (2013). Toward multiscale simulation of moist flows with soundproof equations. *J. Atmospher. Sci.* 70, 3995–4011. doi: 10.1175/JAS-D-13-024.1
- Mattioni, M., and Le Novère, N. (2013). Integration of biochemical and electrical signaling-multiscale model of the medium spiny neuron of the striatum. *PLoS ONE* 8:e66811. doi: 10.1371/journal.pone.0066811
- Pospischil, M., Toledo-Rodriguez, M., Monier, C., Piwkowska, Z., Bal, T., Frégnac, Y., et al. (2008). Minimal hodgkin–huxley type models for different classes of cortical and thalamic neurons. *Biol. Cybern.* 99, 427–441. doi: 10.1007/s00422-008-0263-8
- Rall, W. (1964). “Theoretical significance of dendritic trees for neuronal input-output relations,” in *Neural Theory and Modeling*, ed R. F. Reiss (Stanford, CA: Stanford University Press), 73–97.
- Rüdiger, S., Shuai, J., Huisinga, W., Nagaiah, C., Warnecke, G., Parker, I., et al. (2007). Hybrid stochastic and deterministic simulations of calcium blips. *Biophys. J.* 93, 1847–1857. doi: 10.1529/biophysj.106.099879
- Sand, J., and Skelboe, S. (1992). Stability of backward euler multirate methods and convergence of waveform relaxation. *BIT Numer. Math.* 32, 350–366. doi: 10.1007/BF01994887
- Shampine, L. F., and Reichelt, M. W. (1997). The matlab ode suite. *SIAM J. Sci. Comput.* 18, 1–22. doi: 10.1137/S1064827594276424
- Shukla, J. (2009). “Seamless prediction of weather and climate: a new paradigm for modeling and prediction research,” in *Climate Test Bed Joint Seminar Series* (Camp Springs, MD: NOAA), 8.
- Sjö, A. (1999). *Analysis of Computational Algorithms for Linear Multistep Methods*, Vol. 1999. Lund: Lund University.
- Skelboe, S. (1992). Methods for parallel integration of stiff systems of odes. *BIT Numer. Math.* 32, 689–701. doi: 10.1007/BF01994851
- Skelboe, S. (2000). Accuracy of decoupled implicit integration formulas. *SIAM J. Sci. Comput.* 21, 2206–2224. doi: 10.1137/S1064827598337919
- Söderlind, G., and Wang, L. (2006). Adaptive time-stepping and computational stability. *J. Comput. Appl. Math.* 185, 225–243. doi: 10.1016/j.cam.2005.03.008
- Wilson, M. A., Bhalla, U. S., Uhley, J. D., and Bower, J. M. (1990). “Genesis: a system for simulating neural networks,” in *Advances in Neural Information Processing Systems*, Vol. 89, ed D. Touretzky (San Mateo, CA: Morgan Kaufmann), 485–492.

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2016 Brocke, Bhalla, Djurfeldt, Hellgren Kotaleski and Hanke. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) or licensor are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.