Mathematical Biosciences 263 (2015) 1–17

Contents lists available at ScienceDirect

Mathematical Biosciences

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

Time-stepping techniques to enable the simulation of bursting behavior in a physiologically realistic computational islet

Samuel Khuvis, Matthias K. Gobbert, Bradford E. Peercy*

Department of Mathematics and Statistics, University of Maryland, Baltimore County, 1000 Hilltop Circle, Baltimore, MD 21250, USA

ARTICLE INFO

Article history:
Received 20 May 2014
Revised 3 February 2015
Accepted 5 February 2015
Available online 14 February 2015

Keywords:
Computational islet
Beta cells
Stiff ordinary differential equations
Numerical differentiation formulas
Automatic differentiation
2010 MSC:
37M05
65L04
68W30
92-08
92B25

ABSTRACT

Physiologically realistic simulations of computational islets of beta cells require the long-time solution of several thousands of coupled ordinary differential equations (ODEs), resulting from the combination of several ODEs in each cell and realistic numbers of several hundreds of cells in an islet. For a reliable and accurate solution of complex nonlinear models up to the desired final times on the scale of several bursting periods, an appropriate ODE solver designed for stiff problems is eventually a necessity, since other solvers may not be able to handle the problem or are exceedingly inefficient. But stiff solvers are potentially significantly harder to use, since their algorithms require at least an approximation of the Jacobian matrix. For sophisticated models, systems of several complex ODEs in each cell, it is practically unworkable to differentiate these intricate nonlinear systems analytically and to manually program the resulting Jacobian matrix in computer code. This paper demonstrates that automatic differentiation can be used to obtain code for the Jacobian directly from code for the ODE system, which allows a full accounting for the sophisticated model equations. This technique is also feasible in source-code languages Fortran and C, and the conclusions apply to a wide range of systems of coupled, nonlinear reaction equations. However, when we combine an appropriately supplied Jacobian with slightly modified memory management in the ODE solver, simulations on the realistic scale of one thousand cells in the islet become possible that are several orders of magnitude faster than the original solver in the software Matlab, a language that is particularly user friendly for programming complicated model equations. We use the efficient simulator to analyze electrical bursting and show non-monotonic average burst period between fast and slow cells for increasing coupling strengths. We also find that interestingly, the arrangement of the connected fast and slow heterogeneous cells impacts the peak bursting period monotonically.

1. Introduction

The endocrine system of the pancreas contains clusters of cells called islets of Langerhans, which consist primarily of four different types of cells [15]. The most common type of cell is the β-cell, which is responsible for the secretion of insulin. Since diabetes is characterized by irregular levels of insulin, we are interested in developing a numerical framework that will allow us to calculate and better understand emergent dynamics that lead to the secretion of insulin. Many models of β-cells have been developed that range in focus from electrical bursting and exocytosis to glucose metabolism and have been recently reviewed in Ref. [1]. Furthermore, models of networks of β-cells have been developed [13,17,22]. We use a seven variable β-cell model by Bertram and Sherman [2] and a comparable three variable model by Sherman and Rinzel [8,21].

As an example of a physiological question, we analyze in this paper the effect of electrical coupling between heterogeneous cells in differing patterns of connectivity within a computational islet, represented by a three-dimensional $N \times N \times N$ cube of β-cells. On the physiologically realistic scale of $N^3 = 1000$ cells in an islet, N in $N \times N \times N$ ranges up to 10. By concatenating all unknown variables into a vector y of length $7N^3$ for the seven variable and $3N^3$ for the three variable model, the matrix form of the initial value problems for both islet models with N^3 cells can then be written as a system of coupled ordinary differential equations (ODEs)

$$\frac{dy}{dt} = f^{(\text{ode})}(t,y) = f(t,y) + Gy, \quad 0 < t \le t_f, \quad y(0) = y_0. \tag{1}$$

The two terms on the right-hand side distinguish explicitly the nonlinear reactions in $f(t,y)$ and the coupling between cells involving the matrix G. To capture several burst periods, the simulations for the three variable model are required from 0 ms to 200,000 ms and for the seven variable model the simulations from 0 ms to 500,000 ms.

The reaction equations in these models result in systems of ODEs that are referred to as stiff, since the reaction speeds can vary widely,

* Corresponding author. Tel.: +1 410 455 2436; fax: +1 410 455 1066.
E-mail address: bpeercy@umbc.edu (B.E. Peercy).

http://dx.doi.org/10.1016/j.mbs.2015.02.001
0025-5564/

say between voltage dynamics and gating or endoplasmic reticulum calcium dynamics. ODE solvers appropriate for stiff ODEs necessarily require the Jacobian matrix $J^{(\text{ode})}(t, y) = \nabla_y f^{(\text{ode})}(t, y)$ of the system of ODEs. Since the simulations are needed for large final times to fully capture at least several burst periods, the efficient performance of the ODE solver is crucial. This is the focus of this work.

Some other work that takes into account multiple β-cells to create an islet in three spatial dimensions includes Tsaneva-Atanasova and Sherman [24], who consider electrical coupling and coupling in calcium in a manner similar to our work in a $6 \times 6 \times 6$ islet implemented with Runge–Kutta time-stepping in Fortran 95. Other work by Sherman, Xu, and Stokes [22] considers a $5 \times 5 \times 5$ islet to address the impact of intercellular currents during an experiment where a single islet cell is voltage clamped to record local transmembrane currents. Our works [9] and [16] utilize the islet framework described here with the seven variable model and further include coupling in metabolic variables. Recent work by Pu et al. [17] includes the mention of computer performance. They consider a hexagonal lattice of β-cells with up to 1000 cells using LSODE in Fortran 90. They report that solving a 10 variable model with 1000 cells for 2000 s of model time takes 26 h [17, p. 7]. This information gives an indication of the expected computational load involved in simulations of the intended scale of a seven variable model on $N \times N \times N$ cells.

One of the crucial opportunities for a user to influence the performance of a stiff ODE solver is the choice, how to supply the Jacobian information. Choices range from supplying no information, which results in the code computing a numerical approximation of it, to providing a function that returns the values of the matrix $J^{(\text{ode})}(t, y)$ for inputs (t, y) whenever required by the ODE solver. This function is analogous in interface to the function for $f^{(\text{ode})}(t, y)$ that the user has to provide to the solver in any case to specify the ODE problem (1). But the function for the matrix $J^{(\text{ode})}(t, y)$ is much more difficult to supply, since an analytic formula for each derivative component needs to be calculated and then hand-coded by the user. Since this is a tedious and potentially extremely error prone process, automatic differentiation is a tool that was developed to take the Fortran or C code of a function and differentiate it symbolically. This is different than conventional symbolic differentiation in that both input and output of the process are functions in the same source code as the input function, so it is directly usable as code for a Jacobian function for an ODE solver for which one had to write code for the right-hand side function anyway. It is clear that Matlab offers the same opportunity for automatic differentiation as Fortran or C, but it has taken longer for automatic differentiation libraries to become available. We use the software package ADiMat for automatic differentiation in Matlab [4].

This paper demonstrates the potential advantage of using automatic differentiation in particular in Matlab: Matlab is a language that allows readily the correct implementation of fairly complicated nonlinear model equations, for instance for ODEs, and many users prefer it for this reason over other (particularly source code) languages. By automatic differentiation, the programming of a Jacobian is not a limiting factor any more. In particular, once the machinery of the automatic differentiation is set up, it can readily be used again to obtain a new Jacobian, if the model is changed in any way. This is easier than it would be to hand-calculate the Jacobian again.

To analyze the effect of coupling between cells in a computational islet, we consider an islet of β-cells with varying burst rates [14]. The distribution of these cells is not known, therefore we are investigating several possible distributions of slow bursting and fast bursting cells and capturing their emergent behavior, and we introduce a quantitative measure for the heterogeneity. It turns out that the arrangement of the connected fast and slow heterogeneous cells impacts the peak bursting period monotonically. We also observe that as the heterogeneity of an islet structure increases the peak coupling strength decreases. These simulations demonstrate that both the three variable and the seven variable model have analogous dynamics. This justifies the use of the three variable model as a stand-in for the more complex model in the numerical analyses.

The remainder of this paper is structured as follows: Section 2 describes the physiological background and both the three variable and seven variable models in detail; Section 2.4 specifically defines the model of the coupling strength in (1). Section 3 motivates the inclusion of all available numerical methods, specifies our modifications, and describes the use of ADiMat in more detail. Section 4 contains the full results of our physiological studies. Section 5 presents the numerical performance studies for both models that drive home the need for using an appropriate ODE solver designed for stiff problems and that Matlab can also be an extremely efficient computational tool. For instance, enabling physiologically realistic simulations for the duration of several burst periods on an islet with 1000 cells that takes on the order of 10 min instead of 10 h becomes possible. Finally, Section 6 discusses the detailed conclusions that can be drawn from all reported simulations.

2. Physiological models

In this section, we discuss the physiological background in Section 2.1, we specify the full details of the three variable and seven variable models in Sections 2.2 and 2.3, respectively, and we describe cell coupling in an islet in Section 2.4.

2.1. Physiological background

Diabetes mellitus is a disease characterized by a high concentration of glucose in a person's blood stream. The concentration of glucose in the blood is regulated by insulin, a hormone produced by cells in the pancreas. So, if the concentration of glucose in the blood stream is too high, it is caused mainly by either an insulin deficiency or an insulin resistance which means that insulin does not properly interact with cells to signal glucose uptake. Type 1 diabetes corresponds to an insulin deficiency due to an autoimmune attack on insulin-producing β-cells, while Type 2 diabetes is caused by either an insulin resistance or insulin deficiency. With statistics from January 2011 showing that 23.6 million people in the United States suffer from diabetes (Centers for Disease Control and Prevention [5]), being able to model the cells and their interactions which play a large role in diabetes would be valuable.

The pancreas is an organ in the body which is part of both the endocrine system and digestive system. In the endocrine system in the pancreas are clusters of cells called islets of Langerhans. These islets contain α-cells, β-cells, δ-cells, and pancreatic polypeptide (PP) producing cells along with distributed capillaries, with β-cells the most common type of cell in an islet of Langerhans. An islet's production of insulin, the key hormone in blood glucose maintenance released by β-cells, is related to both its metabolic and electrical activities.

The consensus model of stimulus-secretion coupling illustrates how a β-cell responds to glucose entering the cell. In the consensus model, after glucose enters the β-cell through the glucose transporter GLUT2 it is converted into pyruvate through glycolysis and then metabolized inside mitochondria. This process produces adenosine triphosphate (ATP) and cellular energy at the expense of adenosine diphosphate (ADP). The increase in ATP–ADP ratio results in the closing of K_{ATP} channels. This results in the depolarization of the β-cell which allows calcium to enter the cell. The calcium triggers autocatalytic release of more calcium from the endoplasmic reticulum and the exocytosis of insulin containing secretory granules. The insulin causes the blood glucose to return back to basal levels by signaling to cells throughout the body. As the glucose levels drop at the β-cell, ATP–ADP levels also tend to recover, allowing the K_{ATP} channels to open back up. The opening of these channels stops the depolarizing electrical activity [3].