Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 17(2): 1808–1819.
DOI: 10.3934/mbe.2020095
Received: 03 September 2019
Accepted: 03 December 2019
Published: 17 December 2019

Research article

Combination of multi-variable quadratic adaptive algorithm and hybrid operator splitting method for stability against acceleration in the Markov model of sodium ion channels in the ventricular cell model

Ching-Hsing Luo[1,*] **, Xing-Ji Chen**[1] **and Min-Hung Chen**[2,*]

[1] School of Data and Computer Science, Sun Yat-Sen University, Guangzhou 510006, China

[2] Department of Mathematics, National Cheng Kung University, 1 University Road, Tainan 701, Taiwan

* **Correspondence:** Email: luojinx5@mail.sysu.edu.cn, minhung@gmail.com.

Abstract: Markovian model is widely used to study cardiac electrophysiology and drug screening. Due to the stiffness of Markov model for single-cell simulation, it is prone to induce instability by using large time-steps. Hybrid operator splitting (HOS) and uniformization (UNI) methods were devised to solve Markovian models with fixed time-step. Recently, it is shown that these two methods combined with Chen-Chen-Luo's quadratic adaptive algorithm (CCL) can save markedly computation cost with adaptive time-step. However, CCL determines the time-step size solely based on the membrane potential. The voltage changes slowly to increase the step size rapidly, while the values of state variables of Markov sodium channel model still change dramatically. As a result, the system is not stable and the errors of membrane potential and sodium current exceed 5%. To resolve this problem, we propose a multi-variable CCL method (MCCL) in which state occupancies of Markov model are included with membrane potential as the control quadratic parameters to determine the time-step adaptively. Using fixed time-step RK4 as a reference, MCCL combined with HOS solver has 17.2 times speedup performance with allowable errors 0.6% for Wild-Type Na+ channel with 9 states (WT-9) model, and it got 21.1 times speedup performance with allowable errors 3.2% for Wild-Type Na+ channel with 8 states (WT-8) model. It is concluded that MCCL can improve the simulation instability problem induced by a large time-step made with CCL especially for high stiff Markov model under allowable speed tradeoff.

Keywords: computational modeling; ventricular action potential; sodium Channel; stiff Markov model; stability; adaptive algorithm

1. Introduction

Since advanced computing technology makes the computational models of cardiac electrophysiology getting mature, the computational models become an important basis for clinical applications [1, 2]. Moreover, cardiac cells up to an order of one billion and the internal processes of each cell are also quite complex in the human heart [3]. Therefore, the computational speed, accuracy, and stability are key hotspots in whole heart simulation. Firstly, advanced computer parallelization greatly reduces computing cost of tissue-scale cardiac simulations [4–8]. Secondly, the adaptive algorithms of time-steps and space-steps are a powerful tool for improving the accuracy of computation [6, 9, 10]. Thirdly, stiffness is ubiquitous in cardiac cell models, which easily leads to unstable computation [11]. Aiming to solve the stiff model of a single cell, the advanced numerical methods of ODEs system, Exponential Adams-Bashforth integrators [12] and Gating-enhanced implicit-explicit linear multistep methods [13] were proposed. In addition, Sundials CVODE [14] is a useful solver for ODEs system. The backward-differentiation-formula (BDF) method in CVODE with low tolerances can obtain a highly converged solution for stiff problems. Due to linear multistep methods consuming high computation cost for high precision solutions, using one-step methods to study the tradeoff between speedup and stability has been a new hot-spot in recent years.

The stability of stiff Markov model for ion channels is our main concern. In recent years, Markovian model for ion channels has been widely used in cardiac arrhythmia [15–17] and pharmacologic screening for anti-arrhythmic drugs [18, 19]. The introduction of Markov model makes the cell model become quite stiff, resulting in unstable simulation as large time-step is used to accelerate the simulation. Therefore, the confrontation between the computational speed and stability of a single cell is a noticeable problem. To ensure high simulation stability, Johnny Moreira et al. [20] compared UNI with Forward Euler (FE), Rush-Larsen, First-Order Sundnes et al. (SAST1), Second-Order Sundnes et al. (SAST2), and Second-Order Runge-Kutta (RK2) methods and found that UNI could substantially increase the stability of simple first-order (fixed time-step) methods but with very low speedup. Stary et al. [21] proposed the HOS method in comparison to FE and Matrix Rush-Larsen (MRL) methods to show that HOS method could also increase stability with moderate speedup. While, CCL algorithm [22] has been verified to be an efficient method for the stability of the action potential simulation. And our previous work [23] confirmed that CCL algorithm combined with HOS or UNI method could save markedly computation cost and increased the stability of adaptive methods for time-step from 0.001 to 1 millisecond.

For the simulation of cardiac action potential, Marsh et al. [24] proposed that Mixed Root Mean Square(MRMS) errors should be less than 5% to meet the requirement of clinical applications. The simulation results by Chen et al. [23] even with high speed could not meet the accuracy requirement of clinical applications (less than 5% errors). Due to CCL algorithm only adjusts time-steps according to the first and second derivatives of membrane potential, during the period of slow voltage change (from phase 3 to phase 4), the time-step increases rapidly, causing simulation instability of the state variables for the WT-9 model as the state variables still rapidly change their values, resulting in over 5% MRMS errors at simulations of membrane potential and sodium current. For instance, using fixed time-step HOS method as a reference, CCL+HOS solver can get 295 times speedup, but with 22% and 6.9% MRMS errors for potential and sodium current, respectively. The big errors come from large time-step causing the state occupancies of WT-9 model to be simulated unstably at fast

inactivation, closed-inactivation and the close states. To eliminate this problem, we propose a multi-variable CCL method that takes the first and second derivatives of membrane potential and state occupancies as input parameters, and then generates the appropriate time-step based on the quadratic equation solution of time-step.

2. Materials and methods

2.1. Cardiac ventricular cell model

The Markov model of the Wild-Type (WT) sodium channel [18, 25] is integrated into the dynamic Luo-Rudy (LRd) model [26]. To simulate the ventricular action potential, the formula is given by,

$$dV/dt = -(1/C_m)(I_{ion} + I_{st}) \tag{2.1}$$

$$\begin{aligned} I_{ion} = {} & I_{Na} + I_{Na,b} + I_{Ca(L),Na} + I_{ns,Na} + 3I_{NaK} + 3I_{NaCa} + I_K + I_{Ki} + I_{Kp} \\ & + I_{Ca(L),K} - 2I_{NaK} + I_{ns,K} + I_{Ca(L)} + I_{Ca,b} + I_{p(Ca)} - 2I_{NaCa} \end{aligned} \tag{2.2}$$

where I_{st} is an external stimulus current, with time-step 0.01 or 0.001 ms with duration 0.5 ms for WT-9 or WT-8 model, respectively. $I_{Na}, I_{Na,b}, I_{Ca(L),Na}, I_{ns,Na}, 3I_{NaK}, 3I_{NaCa}$ are sodium channel currents. $I_K, I_{Ki}, I_{Kp}, I_{Ca(L),K}, 2I_{NaK}, I_{ns,K}$ are potassium channel currents. $I_{Ca(L)}, I_{Ca,b}, I_{p(Ca)}, 2I_{NaCa}$ are calcium channel currents. For the detailed descriptions, please see LRd model [26].

2.2. Sodium channel models

2.2.1. The Markov model with nine states (WT-9)

The WT-9 Markovian model of the sodium channel includes nine states [25]. Each state is represented by the proportion of channels with respect to the total number of channels called state occupancy. A Markovian model of the WT sodium channel (Figure 1) is composed of one conducting open state (O), three closed states (C_1, C_2, C_3), two closed-inactivation states (IC_3, IC_2), one fast inactivation state(IF) and two intermediate inactivation states (IM_1, IM_2). According to the principle of microscopic reversibility [27], the change in one state occupancy with continuous-time is equal to the net flow for the transition state. This change can be represented by an ordinary differential equation (ODE). The WT-9 Markovian sodium channel has nine equations that form a linear system known as Kolmogorov equations.

$$d\vec{u}/dt = Q(V)\vec{u} \tag{2.3}$$

where $Q(V)$ is the 9×9 matrix of transition rates (af11, af12, af13, af2, af3, af4, af5, bt11, bt12, bt13, bt2, bt3, bt4, bt5). $Q(V)$ is a real matrix of nine rows by nine columns, and it contains elements q_{ij}. $\vec{u}$ is a vector with nine states $[O, C_1, C_2, C_3, IC_3, IC_2, IF, IM_1, IM_2]^T$.

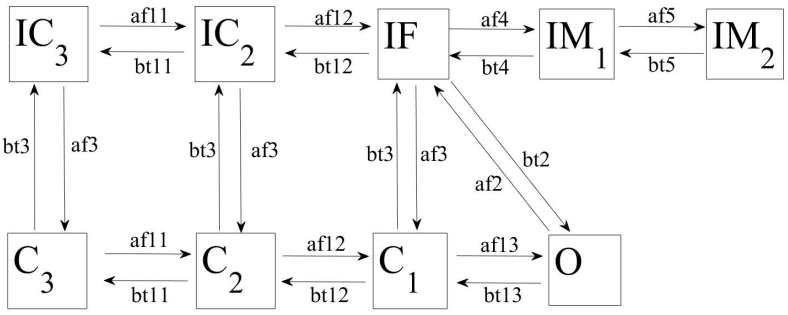

Figure 1. WT-9 sodium channel Markov chain model with nine states.

The formulation of the sodium channel is described by,

$$I_{Na} = \overline{g_{Na}} \times \mathrm{O} \times (V - E_{Na}) \tag{2.4}$$

where $\overline{g_{Na}}$ is the conductance constant of the sodium channel and E_{Na} is the equilibrium potential.

2.2.2. The Markov model with eight states (WT-8)

A WT-8 Markovian model with eight states of the sodium channel [18] is composed of one conducting open state (O), three closed states ($\mathrm{C}_1, \mathrm{C}_2, \mathrm{C}_3$), two closed-inactivation states ($\mathrm{IC}_3, \mathrm{IC}_2$), one fast inactivation state(IF) and slow inactivation state (IS) shown in Figure 2.

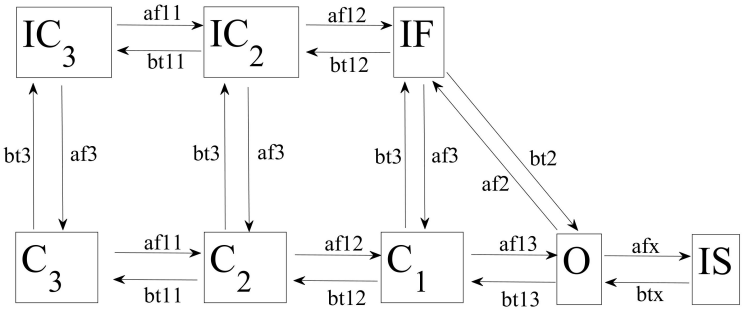

Figure 2. WT-8 sodium channel Markov chain model with eight states.

$Q(V)$ in Eq (2.3) is the 8×8 matrix of transition rates (af11, af12, af13, af2, af3, afx, bt11, bt12, bt13, bt2, bt3, btx). $Q(V)$ is a real matrix of eight rows by eight columns, and it contains elements q_{ij}. The vector $\overrightarrow{u}$ includes eight states $[\mathrm{O}, \mathrm{C}_1, \mathrm{C}_2, \mathrm{C}_3, \mathrm{IC}_3, \mathrm{IC}_2, \mathrm{IF}, \mathrm{IS}]^T$.

2.3. The numerical schemes

2.3.1. The CCL method

CCL method is applied to change time-step with membrane potential in simulations. The maximum time-step $\Delta t_{max} = 1$ ms and the minimum time-step $\Delta t_{min} = 0.001$ ms. Time-step is changed in the scope of $\Delta t_{min} \sim \Delta t_{max}$. To create a fine adaptive time-step, the new approximated membrane potential with second-order Taylor expansion is given by,$V(t_{n+1}) \cong V(t_n) + dV/dt \times \Delta t + 1/2 \times d^2V/dt^2 \times \Delta t^2$. Which can be a quadratic Δt equation used to update time-steps adaptively. For a detailed description, please see Chen et al. [22]. The equation is given by,

$$1/2 \times a \times \Delta t^2 + b \times \Delta t - c \cong 0, \text{where } a = d^2V/dt^2, b = dV/dt, c = V(t_{n+1}) - V(t_n) = 0.1 \text{ mV} \quad (2.5)$$

Let $D = b^2 + 4 \times a/2 \times c$, The solutions of Eq (2.5) are shown,

$$\Delta t = (-b + \sqrt{D})/a, \ if\ b \geq 0, D \geq 0, \text{ where } \Delta t \in (0.001 \sim 1) \text{ ms}, c = 0.1 \text{ mV} \quad (2.6)$$

$$\Delta t = (-b - \sqrt{D})/a, \ if\ b < 0, D \geq 0, \text{ where } \Delta t \in (0.001 \sim 1) \text{ ms}, c = -0.1 \text{ mV} \quad (2.7)$$

$$\Delta t = -b/a, \ if\ D < 0, \text{ where } \Delta t \in (0.001 \sim 1) \text{ ms} \quad (2.8)$$

$$\Delta t_{new} = Minimum(\Delta t, \Delta t_{max}), \text{ where } \Delta t_{max} = Minimum(2 \times \Delta t_{old}, \Delta t_{max}) \quad (2.9)$$

2.3.2. The multi-variable CCL method

In addition to membrane potential, the variables control in MCCL method is expanded to multiple variables given by,

$$1/2 \times a \times \Delta t_i^2 + b \times \Delta t_i - c \cong 0, \text{ where } a = d^2X_i/dt^2, b = dX_i/dt \quad (2.10)$$

c is the offset of variable X_i, X_i is one of the variables $V_n, \mathrm{O}, \mathrm{C}_1, \mathrm{C}_2, \mathrm{C}_3, \mathrm{IC}_3, \mathrm{IC}_2, \mathrm{IF}, \mathrm{IS}, \mathrm{IM}_1, \mathrm{IM}_2$ at i^{th} time. V_n is the normalized membrane potential given by, $V_n = (V - V_{rest})/N_V$, where $N_V = |V_{rest} - V_{reversal}|$. V_{rest} is the membrane potential of rest state, $V_{reversal}$ is the reversal potential of sodium channel. $V_{rest} = -88.65$ mV and $V_{reversal} = 70.50$ mV are set respectively. Let $c = X_i(t_{n+1}) - X_i(t_n) = 0.1/N_V$ and $D = b^2 + 4 \times a/2 \times c$, then The solutions of Eq (2.10) are shown ,

$$\Delta t_i = (-b + \sqrt{D})/a, \ if\ b \geq 0, D \geq 0, \text{ where } \Delta t_i \in (\Delta t_{min} \sim \Delta t_{max}) \text{ ms}, c = 0.1/N_V \quad (2.11)$$

$$\Delta t_i = (-b - \sqrt{D})/a, \ if\ b < 0, D \geq 0, \text{ where } \Delta t_i \in (\Delta t_{min} \sim \Delta t_{max}) \text{ ms}, c = -0.1/N_V \quad (2.12)$$

$$\Delta t_i = -b/a, \ if\ D < 0, \text{ where } \Delta t_i \in (\Delta t_{min} \sim \Delta t_{max}) \text{ ms} \quad (2.13)$$

$$\Delta t_i = Minimum(\Delta t_i, \Delta t_{max}), \text{ where } \Delta t_{max} = Minimum(2 \times \Delta t_{old}, \Delta t_{max}) \quad (2.14)$$

For WT-9 model,

$$\Delta t_{new} = Minimum(\Delta t_{V_n}, \Delta t_{\mathrm{O}}, \Delta t_{\mathrm{C}_1}, \Delta t_{\mathrm{C}_2}, \Delta t_{\mathrm{C}_3}, \Delta t_{\mathrm{IC}_3}, \Delta t_{\mathrm{IC}_2}, \Delta t_{\mathrm{IF}}, \Delta t_{\mathrm{IM}_1}, \Delta t_{\mathrm{IM}_2}) \quad (2.15)$$

Or, for WT-8 model,

$$\Delta t_{new} = Minimum(\Delta t_{V_n}, \Delta t_{\mathrm{O}}, \Delta t_{\mathrm{C}_1}, \Delta t_{\mathrm{C}_2}, \Delta t_{\mathrm{C}_3}, \Delta t_{\mathrm{IC}_3}, \Delta t_{\mathrm{IC}_2}, \Delta t_{\mathrm{IF}}, \Delta t_{\mathrm{IS}}) \quad (2.16)$$

2.3.3. The Markovian model solvers

Four methods [23] are mainly adopted to solve transient solutions of Eq (2.3) as shown below,
1) Forward Euler method ("FE") and FE method is given by,

$$\overrightarrow{u}(t_n + \Delta t) = \overrightarrow{u}(t_n) + \Delta t Q(V(t_n))\overrightarrow{u}(t_n) \tag{2.17}$$

2) Taylor series expansion ("TAL") and second-order Taylor series method is given by,

$$\overrightarrow{u}(t_n + \Delta t) = \overrightarrow{u}(t_n) + \frac{\Delta t Q(V(t_n))}{1!}\overrightarrow{u}(t_n) + \frac{(\Delta t Q(V(t_n)))^2}{2!}\overrightarrow{u}(t_n) \tag{2.18}$$

3) Uniformization ("UNI") and the uniformization method by Sidje et al. [28] is given by,

$$\begin{aligned}\overrightarrow{u}(t_n + \Delta t) &= \exp(\Delta t Q(V(t_n)))\overrightarrow{u}(t_n) \\ &\approx \overrightarrow{u}(t_n) + \frac{\Delta t Q(V(t_n))}{1!}\overrightarrow{u}(t_n) + \cdots + \frac{(\Delta t Q(V(t_n)))^l}{l!}\overrightarrow{u}(t_n)\end{aligned} \tag{2.19}$$

$$\begin{aligned}\overrightarrow{u}(t_n + \Delta t) &= \exp(q\Delta t Q(V(t_n))^* - I)\overrightarrow{u}(t_n) \\ &= \exp(-q\Delta t)\exp(q\Delta t Q(V(t_n))^*)\overrightarrow{u}(t_n)\end{aligned} \tag{2.20}$$

$$\bar{\overrightarrow{u}}(t_n + \Delta t) = \sum_{k=0}^{l} \exp(-q\Delta t)\frac{(q\Delta t)^k}{k!}(Q(V(t_n))^*)^k\overrightarrow{u}(t_n) \tag{2.21}$$

$$\sum_{k=0}^{l} \exp(-q\Delta t)\frac{(q\Delta t)^k}{k!} \geq 1 - \varepsilon_{tol} \tag{2.22}$$

$$\bar{\overrightarrow{u}}(t_n + \Delta t) = (\exp(-q\Delta t)\exp(q\Delta t Q(V(t_n))^*))^m\overrightarrow{u}(t_n) \tag{2.23}$$

where $m = \lceil \frac{q\Delta t}{\theta} \rceil, h = \frac{\Delta t}{m}, Q(V(t_n))^* = I + \frac{Q(V(t_n))}{q}, q = \max|q_{ii}|$. q is the maximum diagonal element, in absolute value, of matrix Q. ε_{tol} denotes the prescribed error tolerance, θ and m are parameters to partition the integral domain for overflow prevention. Two parameters are set in advance, the default ε_{tol} is set 10^{-11} and $\theta = 1$ unless otherwise specified. The truncation point l is determined by Eq (2.22).

4) Hybrid Operator Splitting ("HOS") and HOS method is given by,

$$Q(V(t_n)) = Q_0(V(t_n)) + Q_1(V(t_n)) + Q_2(V(t_n)) \tag{2.24}$$

$$\overrightarrow{u}_{n+1/3} = \exp(\Delta t Q_0(V(t_n)))\overrightarrow{u}_n \tag{2.25}$$

$$\overrightarrow{u}_{n+2/3} = \exp(\Delta t Q_1(V(t_n)))\overrightarrow{u}_{n+1/3} \tag{2.26}$$

$$\overrightarrow{u}_{n+1} = \exp(\Delta t Q_2(V(t_n)))\overrightarrow{u}_{n+2/3} \tag{2.27}$$

For a detailed description, please see Stary et al. [21].

2.4. Computational simulations

2.4.1. Numerical schemes

To compare the performance and accuracy between CCL and MCCL methods combined with Markov solver, we have tested eight hybrid solvers, in which each of four methods (FE, TAL, UNI,

Table 1. Numerical schemes.

Solver	Time-steps	Solver	Time-steps
CCL+FE	(0.001~1 ms)	MCCL+FE	(0.001~1 ms)
CCL+TAL	(0.001~1 ms)	MCCL+TAL	(0.001~1 ms)
CCL+UNI	(0.001~1 ms)	MCCL+UNI	(0.001~1 ms)
CCL+HOS	(0.001~1 ms)	MCCL+HOS	(0.001~1 ms)

HOS) is combined with CCL or MCCL method as listed in Table 1. The reference solver is fourth-order four-stage Runge-Kutta (RK4) method used to solve Markov model with fixed time-step (Δt = 0.001 ms). For the very sharply stiff model, BDF (backward-differentiation formula) with tight tolerance and fixed time-step 0.001 ms in Sundials CVODE as a benchmark is an alternative good choice due to high stability even with high computation cost. RK4 with fixed step 0.001 ms has less than 0.6% MRMSE deviation and 10 times smaller computation cost in comparison to BDF with tight tolerance and fixed time-step 0.001 ms in Sundials CVODE.

2.4.2. Computing platform and evaluated errors

We implement the numerical experiments with Visual Studio 2013 C/C++ compiler. All the numerical tests are performed on a desktop computer equipped with an Intel(R) Core (TM) i7-7700 CPU running 64-bit Windows10 × 64. The fixed time-step scheme with RK4 is used as reference solution to evaluate the performance of adaptive time-step schemes with Markov solvers. The Mixed Root Mean Square Error (MRMSE) [24] is used as a metric of quantitative error. It is given by,

$$MRMSE \doteq \sqrt{\sum_{i=1}^{N} \frac{\left(\frac{X_i - \hat{X}_i}{1+|\hat{X}_i|}\right)^2}{N}} \tag{2.28}$$

Where X_i is the numerical solution, $\hat{X}_i$ is the reference solution, N is the number of temporal points. All of the numerical schemes run 3000 basic cycle length (BCL) or beats with 1000 ms per BCL or beat. The simulation data are taken from the 1^{st}, 500^{th}, 1000^{th}, 1500^{th}, 2500^{th}, and 3000^{th} beats.

3. Results

3.1. WT-9 Markov model

Table 2 shows the MRMSE and computational performance of the eight schemes for the WT-9 model.

To meet the requirement of clinical applications, Marsh et al. [24] suggested that MRMSE of simulations should be less than 5%. Table 2 shows that all of CCL combined any Markov solver (FE, TAL, UNI, HOS) do not satisfy the requirement, nor MCCL+FE or MCCL+TAL solver. Only MCCL+UNI and MCCL+HOS solvers can meet the 5% requirement. Particularly, MCCL+HOS solver has less than 1% MRMSEs in voltage and sodium current with 17.2 times (17.2X) speedup which is greater than 14.8X for MCCL+UNI solver. It is suggested that the optimal solver is MCCL+HOS solver among eight solvers in Table 2 for the WT-9 model.

Table 2. MRMSE and computational performance for WT-9 model.

Solvers	MRMSE-V	MRMSE-I	Speedup	Cost(ms)
DT+RK4	0.0	0.0	1.0X	$1.0 \times 10^7 \pm 9.8 \times 10^4$
CCL+FE	0.016	0.588	109.9X	$9.5 \times 10^4 \pm 4.0 \times 10^2$
CCL+TAL	0.745	1.158	33.0X	$3.2 \times 10^5 \pm 1.2 \times 10^3$
CCL+UNI	0.056	0.012	16.4X	$6.4 \times 10^5 \pm 8.4 \times 10^2$
CCL+HOS	0.236	0.064	412.3X	$2.6 \times 10^4 \pm 3.2 \times 10^2$
MCCL+FE	0.022	0.065	96.9X	$1.1 \times 10^5 \pm 2.7 \times 10^2$
MCCL+TAL	0.015	0.124	29.8X	$3.5 \times 10^5 \pm 1.4 \times 10^3$
MCCL+UNI	0.021	0.009	14.8X	$7.0 \times 10^5 \pm 3.4 \times 10^3$
MCCL+HOS	0.006	0.002	17.2X	$6.1 \times 10^5 \pm 3.7 \times 10^3$

Note: MRMSE-V: MRMSE of voltage in comparison to the reference. MRMSE-I: MRMSE of sodium current in comparison to the reference. Speedup: The computation cost ratio of one of eight schemes to the reference solution DT+RK4. X, times speedup. Ten simulations for each solver. Cost in Mean ± Standard deviation.

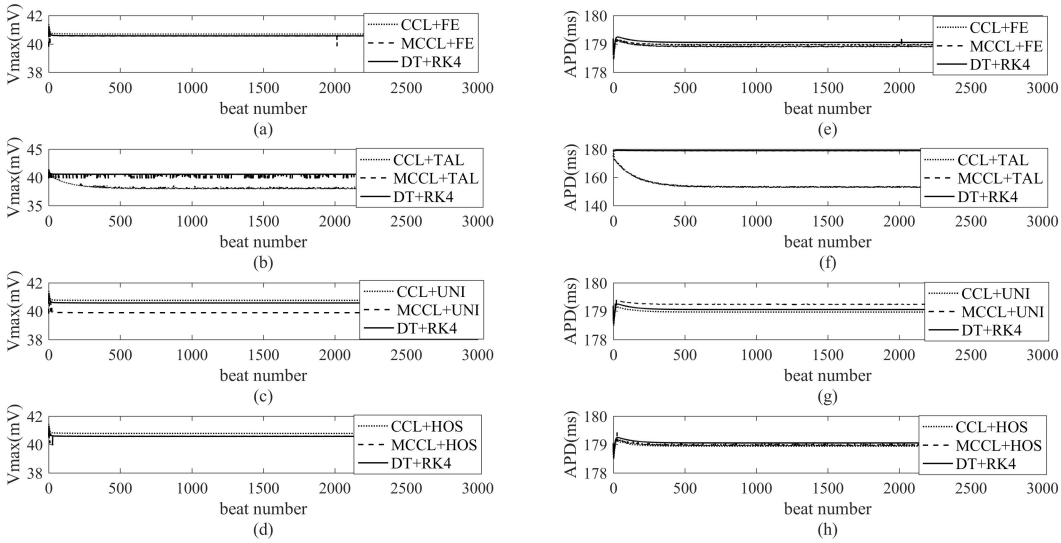

Figure 3. Vmax and APD over 3000 beats using eight solvers and reference solver for WT-9 model.

Figure 3 shows the long-term stability (3000 beats) of CCL method or MCCL method combined Markov solvers for Vmax and APD. Figure 3a,e show that both Vmax and APD of MCCL+FE solver have some unstable fine fluctuation, but they are closer to the reference solution than those of CCL+FE solver. Figure 3b,f show that Vmax and APD of MCCL+TAL solver are close to the reference solution with tiny unstable fluctuations, while Vmax and APD of CCL+TAL solver not only have some tiny unstable oscillation but also deviates far from the reference solution. Importantly, Figure 3c,g show that Vmax of CCL+UNI and that of MCCL+UNI solver have a slight error, but remain stable. Figure 3d,h show that the Vmax and APD of MCCL+HOS solver are closer to the

reference solution than that of CCL+HOS solver. It is indicated that MCCL method can guarantee long-term stability and obtain more accurate solutions about Vmax and APD than CCL method.

Figure 4 shows the occupancy of nine states of Markov model for CCL or MCCL combined HOS solver at first beat. Figure 4a,c,f–i show that all these state occupancies ($O, C_1, C_3, IC_3, IM_1, IM_2$) are close to the reference solution very well for CCL+HOS and MCCL+HOS solvers. While, in Figure 4b shows that the state occupancy IF of MCCL+HOS solver has tiny unstable fluctuation (inset in Figure 4b), but IF of CCL+HOS solver shows a small deviation near 200 ms. Similarly, in Figure 4d,e state C_2 and IC_2 occupancies of MCCL+HOS solver only have very minor fluctuation (insets in Figure 4d,e), while that of CCL+HOS solver presents a deviation error with rapidly descending towards abscissa near 200 ms. It is suggested that MCCL+HOS solver enhances the instability of CCL+HOS solver.

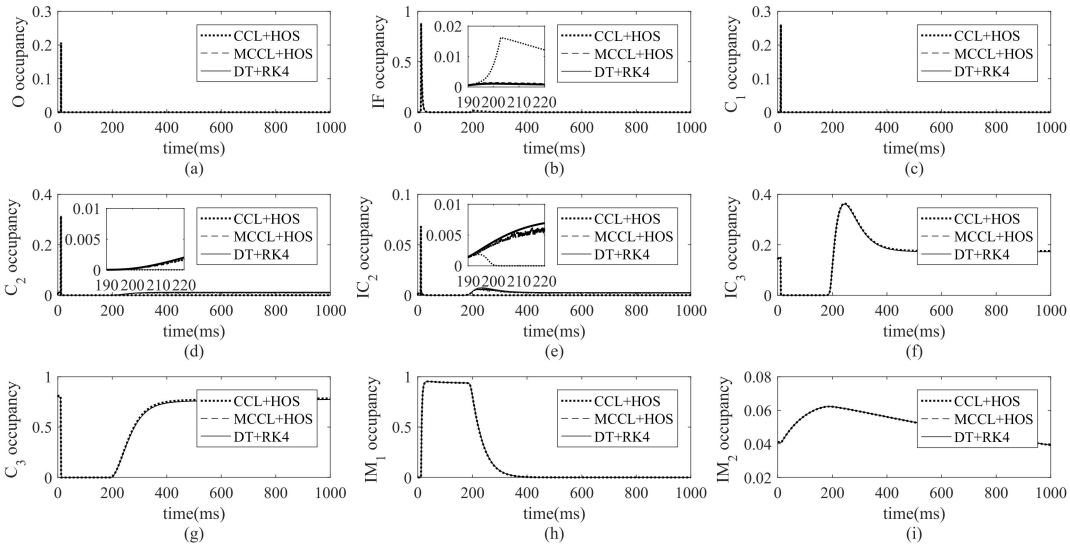

Figure 4. Occupancy of nine states of Markov model for HOS solver at 1^{st} beat for WT-9 model. The insets in b,d,e are the enlarged view from 190 to 220 ms.

Table 3. MRMSE and computational performance for WT-8 model.

Solvers	MRMSE-V	MRMSE-I	Speedup	Cost(ms)
DT+RK4	0.0	0.0	1.0X	$1.1 \times 10^7 \pm 6.0 \times 10^4$
CCL+FE	- -	- -	- -	failure
CCL+TAL	- -	- -	- -	failure
CCL+UNI	0.019	0.008	1.2X	$8.9 \times 10^6 \pm 1.8 \times 10^4$
CCL+HOS	0.250	0.114	266.6X	$3.3 \times 10^4 \pm 1.3 \times 10^2$
MCCL+FE	0.103	3.345	19.8X	$5.3 \times 10^5 \pm 2.5 \times 10^3$
MCCL+TAL	- -	- -	- -	failure
MCCL+UNI	0.076	0.042	1.2X	$8.9 \times 10^6 \pm 1.1 \times 10^4$
MCCL+HOS	0.032	0.013	21.1X	$5.0 \times 10^5 \pm 1.8 \times 10^3$

3.2. WT-8 Markov model

Table 3 shows the MRMSE and computational performance of the eight simulate schemes for the WT-8 model.

CCL+FE, CCL+TAL and MCCL+TAL solvers are the failure, while MCCL+FE solver is success but with MRMSE higher than 5%. Only CCL+UNI and MCCL+HOS solvers have MRMSE-V and MRMSE-I less than 5%. CCL+UNI solver almost takes no speedup, while MCCL+HOS solver still obtains 21.1 times speedup. For WT-8 model, MCCL+HOS solver is the best solver among eight solvers in Table 3.

4. Conclusions

Due to an enormous maximum time-step size, Δt_{max}, used by CCL and MCCL algorithms, the simulation is prone to instability and even divergence to failure for CCL and MCCL combined with FE or TAL, especially for WT-8 model. It is improper to use FE or TAL combined with adaptive time-step methods unless a smaller maximum time-step size, Δt_{max}, is used but then the adaptivity of the method is limited.

In Tables 2 and 3, the speedup performance of MCCL+UNI and CCL+UNI solvers is similar since the UNI algorithm consumes most of the computation time. The formula (2.23) is iterated 260 times with $q = 260$ for the WT-8 model to reach the preset tolerant error even time-step is set as 1ms. As a result, there is almost no speedup of MCCL or CCL combined with UNI for WT-8 model.

While, the speedup of MCCL+HOS and CCL+HOS solvers is significant due to HOS being lightweight. CCL+HOS solver gets high over 250 times speedup for WT-8 and WT-9 models while MCCL+HOS only gets about 20 times. However, CCL+HOS solver has large MRMSE for WT-9 model and even 25% MRMSE for WT-8 failing the clinical application requirement. While MCCL+HOS solver has MRMSEs less than 5% clinical application requirement even with lower speedup.

In our previous work [23], we find that CCL+UNI solver is the most stable but very time-consuming while CCL+HOS solver is less stable but more efficient. It is suggested the optimal solver should take tradeoff between stability and acceleration. In this work, the MCCL+HOS solver is more stable, more accurate and faster than MCCL+UNI solver. For the WT-8 model, CCL+UNI solver is the most accurate but has poor speedup performance (only 1.2X). It is concluded that MCCL+HOS solver is a good solver, especially for high stiff Markov model such as WT-8 sodium channel and MCCL instead of CCL can be a good speedup algorithm tradeoff for stability against acceleration.

Both CCL and MCCL have their limitations in which the speedup of MCCL+HOS solver is far less than CCL+HOS while the accuracy of CCL+HOS is far worse than MCCL+HOS. To achieve better accuracy, MCCL+HOS is highly recommended but its speedup performance is only moderate and, importantly, the speedup performance is about the same for moderate stiff WT-9 and high stiff WT-8 models. It is highly recommended to use MCCL+HOS for a high stiff model, but for moderate stiff model MCCL+HOS should be revised to gain higher speedup for the logic of lower stiff higher speedup performance in the near future.

Acknowledgments

This work was supported by Sun Yat-sen University, China, under Scientific Initiation Project [No.67000-18821109] for High-level Experts and by the Ministry of Science and Technology, Taiwan, under grant 107-2115-M-006-013 separately.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. A. Lopezperez, R. Sebastian, J. M. Ferrero, Three-dimensional cardiac computational modelling: Methods, features and applications, *Biomed. Eng. Online*, **14** (2015), 35.
2. P. Pathmanathan, R. A. Gray, Validation and trustworthiness of multiscale models of cardiac electrophysiology, *Front. Physiol.*, **9** (2018), 106.
3. C. P. Adler, U. Costabel, Cell number in human heart in atrophy, hypertrophy, and under the influence of cytostatics, *Recent Adv. Stud. Card. Struct. Metab.*, **6** (1975), 343–355.
4. Y. Xia, K. Wang, H. Zhang, Parallel optimization of 3d cardiac electrophysiological model using gpu, *Comput. Math. Methods Med.*, **2015** (2015), 862735.
5. J. Langguth, L. Qiang, N. Gaur, C. Xing, Accelerating detailed tissue-scale 3d cardiac simulations using heterogeneous cpu-xeon phi computing, *Int. J. Parallel Program.*, **45** (2016), 1–23.
6. R. Sachetto Oliveira, B. Martins Rocha, D. Burgarelli, W. Meira, C. Constantinides, R. Weber Dossantos, Performance evaluation of gpu parallelization, space-time adaptive algorithms, and their combination for simulating cardiac electrophysiology, *Int. J. Numer. Method Biomed. Eng.*, **34** (2017), e2913.
7. N. Altanaite, J. Langguth, Gpu-based acceleration of detailed tissue-scale cardiac simulations, in *Proceedings of the 11th Workshop on General Purpose GPUs*, ACM, 2018, 31–38.
8. E. Esmaili, A. Akoglu, S. Hariri, T. Moukabary, Implementation of scalable bidomain-based 3d cardiac simulations on a graphics processing unit cluster, *J. Supercomput.*, **75** (2019), 1–32.
9. V. M. Garcia-Molla, A. Liberos, A. Vidal, M. S. Guillem, J. Millet, A. Gonzalez, et al., Adaptive step ode algorithms for the 3d simulation of electric heart activity with graphics processing units, *Comput. Biol. Med*, **44** (2014), 15–26.
10. N. Chamakuri, Parallel and space-time adaptivity for the numerical simulation of cardiac action potentials, *Appl. Math. Comput.*, **353** (2019), 406–417.
11. R. J. Spiteri, R. C. Dean, Stiffness analysis of cardiac electrophysiological models, *Annals Biomed. Eng.*, **38** (2010), 3592.
12. Y. Coudire, C. Douanla-Lontsi, C. Pierre, Exponential adams?bashforth integrators for stiff odes, application to cardiac electrophysiology, *Math. Comput. Simul.*, **153** (2018), 15–34.
13. K. R. Green, R. J. Spiteri, Gating-enhanced imex splitting methods for cardiac monodomain simulation, *Numer. Algorithms*, **81** (2019), 1443–1457.

14. A. C. Hindmarsh, R. Serban, D. R. Reynolds, Sundials: Suite of nonlinear and differential/algebraic equation solvers. Available from: https://computing.llnl.gov/projects/sundials/sundials-software.
15. J. R. Bankston, K. J. Sampson, S. Kateriya, I. W. Glaaser, D. L. Malito, W. K. Chung, et al., A novel lqt-3 mutation disrupts an inactivation gate complex with distinct rate-dependent phenotypic consequences, *Channels*, **1** (2007), 273–280.
16. A. Greer-Short, S. A. George, S. Poelzing, S. H. Weinberg, Revealing the concealed nature of long-qt type 3 syndrome, *Circ.: Arrhythmia Electrophysiol.*, **10** (2017), e004400.
17. C. Campana, I. Gando, R. B. Tan, F. Cecchin, W. A. Coetzee, E. A. Sobie, Population-based mathematical modeling to deduce disease-causing cardiac Na+ channel gating defects, *Biophys. J.*, **114** (2018), 634–635.
18. J. D. Moreno, T. J. Lewis, C. E. Clancy, Parameterization forin-silicomodeling of ion channel interactions with drugs, *Plos One*, **11** (2016), e0150761.
19. A. Tveito, M. Maleckar Mary, T. Lines Glenn, Computing optimal properties of drugs using mathematical models of single channel dynamics, *Comput. Math. Biophys.*, **6** (2018), 41.
20. J. M. Gomes, A. Alvarenga, R. S. Campos, B. M. Rocha, A. P. C. da Silva, R. W. dos Santos, Uniformization method for solving cardiac electrophysiology models based on the markov-chain formulation, *IEEE Trans. Biomed. Eng.*, **62** (2015), 600–608.
21. T. Stary, V. N. Biktashev, Exponential integrators for a markov chain model of the fast sodium channel of cardiomyocytes, *IEEE Trans. Biomed. Eng.*, **62** (2015), 1070–1076.
22. M. H. Chen, P. Y. Chen, C. H. Luo, Quadratic adaptive algorithm for solving cardiac action potential models, *Comput. Biol. Med.*, **77** (2016), 261–273.
23. X. J. Chen, C. H. Luo, M. H. Chen, X. Zhou, Combination of "quadratic adaptive algorithm" and "hybrid operator splitting" or uniformization algorithms for stability against acceleration in the markov model of sodium ion channels in the ventricular cell model, *Med. Biol. Eng. Comput.*, **57** (2019), 1367–1379.
24. M. E. Marsh, S. T. Ziaratgahi, R. J. Spiteri, The secrets to the success of the rush-larsen method and its generalizations, *IEEE Trans. Biomed. Eng.*, **59** (2012), 2506–2515.
25. C. E. Clancy, R. Yoram, Na(+) channel mutation that causes both brugada and long-qt syndrome phenotypes: A simulation study of mechanism, *Circulation*, **105** (2002), 1208–1213.
26. C. H. Luo, Y. Rudy, A dynamic model of the cardiac ventricular action potential. i. simulations of ionic currents and concentration changes, *Circ. Res.*, **74** (1994), 1071–1096.
27. B. Hille, *Ion channels of excitable membranes*, 3rd edition, Sinauer Sunderland, MA, 2001.
28. R. B. Sidje, K. Burrage, S. MacNamara, Inexact uniformization method for computing transient distributions of markov chains, *SIAM J. Sci. Comput.*, **29** (2007), 2562–2580.

AIMS Press