Received: 22 May 2019 | Accepted: 3 June 2019

DOI: 10.1002/pamm.201900295

Simulation of Calcium Waves in a Heart Cell on Modern Multi-Core Parallel Computing Platforms

Carlos Barajas[1], **Stefan Kopecz**[2], **Andreas Meister**[2], **Bradford E. Peercy**[1], and **Matthias K. Gobbert**[1,*]

[1] Department of Mathematics and Statistics, University of Maryland Baltimore County, Baltimore, MD 21250, USA
[2] Institut für Mathematik, Universität Kassel, Heinrich-Plett-Str. 40, 34132 Kassel, Germany

Scalability of parallel computer code is a concern in applications that rely on efficient simulations. We present a demonstration of weak scalability of a hybrid MPI+OpenMP code for the numerical solution of a system of seven coupled, non-linear, time-dependent reaction-diffusion equations that model calcium induced calcium release in a heart cell. The results demonstrate that care needs to be taken in the implementation of the OpenMP pragmas to achieve weak scalability within a shared-memory node and that MPI is capable to provide strong scalability across multiple nodes in a distributed-memory cluster.

1 Introduction

Scalability of parallel computer code is a concern in applications that rely on efficient simulations. Modern multi-core CPUs can have 16 or more computational cores, and a two-socket node thus allows for parallelism with 32 or more process threads per node. To pool the resources of multiple nodes in a distributed-memory cluster requires the use of MPI, but the parallelism within a shared-memory node can be accessed by a combination of MPI processes and OpenMP threads. We present here a demonstration of weak scalability of a hybrid MPI+OpenMP code for the numerical solution of a system of coupled, non-linear, time-dependent reaction-diffusion equations

$$u_t^{(i)} - \nabla \cdot \left(D^{(i)} \nabla u^{(i)} \right) = q^{(i)}(u^{(1)}, \ldots, u^{(n_s)}, \mathbf{x}, t) \quad i = 1, \ldots, n_s, \tag{1}$$

for concentrations $u^{(i)} = u^{(i)}(\mathbf{x}, t)$, $i = 1, \ldots, n_s$, of space $\mathbf{x} \in \Omega \subset \mathbb{R}^3$ and time $0 \leq t \leq t_{\text{fin}}$. Here, $D^{(i)}$ and $q^{(i)}$ denote the diffusivity matrices and non-linear forcing functions, respectively, of the n_s species.

2 Calcium Induced Calcium Release

The application example is given by a model for calcium waves in a heart cell. The original model in [1, 2] started with three species modeling the self-initiation of waves, which is of interest due to the risk of irregular heart beat. This model is also used in [3] and [4] for numerical tests on the finite volume method as spatial discretization and positivity-preserving Patankar time-stepping schemes, respectively. An extension of the model in [5] uses seven variables $n_s = 7$ in (1) to model interaction with the calcium store in the cell, the electrical system of the cell, and the effect of mechanical contraction. The calcium waves through the cell are sustained by calcium release from the store at discrete locations, the calcium release units (CRUs), that are opened by the increase of calcium concentration at these locations. This effect is termed calcium induced calcium release (CICR) and requires long-time simulations with a final time such as $t_{\text{fin}} = 1,000$ ms to allow the study of multiple waves within the time scale studied. Eventually, the goal is to reach laboratory time scales of yet larger final times such as of several minutes, thus the interest in efficient parallel code for this problem. Simultaneously, the spatial mesh needs to be fine, since the CRUs are point sources, mathematically modeled as Dirac delta distributions, at several thousands of locations. This can be seen in Figure 1, which shows snapshots at several times of dots indicating which CRUs are open at that time on a $15 \times 15 \times 31 = 6{,}975$ lattice of CRUs. With a physiological model of CRUs closing for 100 ms after every opening, the snapshots at 100 ms intervals bring out the repeating nature of the calcium wave that starts originally near the center of the three-dimensional domain and spreads from there repeatedly, forming waves in the left and right directions through the cell.

3 Parallel Performance Study

We use a state-of-the-art computing cluster of 42 nodes with two 18-core Intel Skylake CPUs and 384 GB of memory each, connected by an EDR InfiniBand interconnect. In [6], strong scalability was investigated both across multiple nodes in a distributed-memory cluster as well as when using 1, 2, 4, 8, 16, 32 MPI processes per node. Each of these MPI processes constitutes one parallel thread and thus all but the last case leave a lot of cores of the node unused. This is overcome by hybrid MPI+OpenMP code, where always 32 cores are in use through a combination of p_n MPI processes and t_p OpenMP threads per MPI process such that the product $p_n\, t_p = 32$. Table 1 shows the performance study using a shorter final time

* Corresponding author: e-mail gobbert@umbc.edu, phone +1 410 455 2404, fax +1 410 455 1066

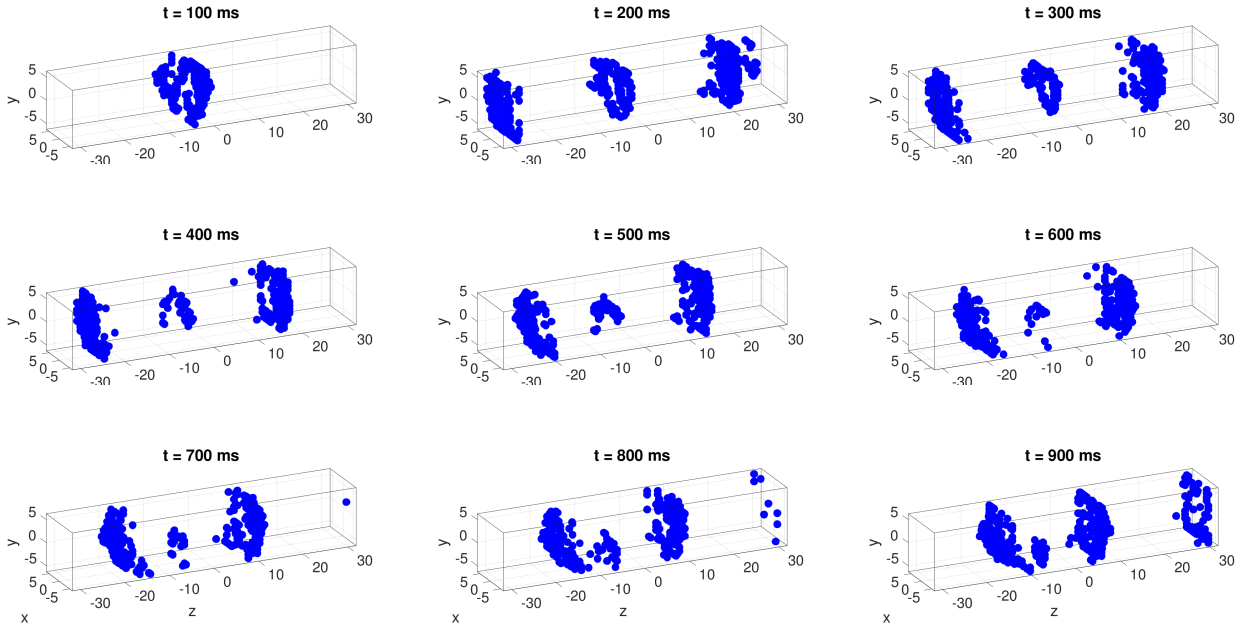

Fig. 1: Snapshots of open calcium release units throughout the cell at times $t = 100, 200, \ldots, 900$ ms.

of $t_{\text{fin}} = 100$ ms for the performance study for two three-dimensional meshes with 983,367 and 7,600,775, respectively, degrees of freedom (DOF) for the seven variable model, i.e., the number of unknowns that have to be computed in each of the 2,667 and 3,295, respectively, time steps. Within each node, this experiment constitutes a weak scalability study, since the same amount of 32 cores per node as resources are used in each row of the table. It becomes apparent that MPI-only code is slightly faster than the hybrid MPI+OpenMP code, which shows that the OpenMP pragmas need to be carefully programmed. When using multiple distributed-memory nodes, excellent speedup is apparent, which shows that MPI parallelism across the high-performance InfiniBand network provides strong scalability.

Table 1: Wall clock time in HH:MM:SS using $N_n = 1, 2, 4$ nodes for seven variable model to final time $t_{\text{fin}} = 100$ ms.

Mesh resolution System dimension		$N_x \times N_y \times N_z = 32 \times 32 \times 128$ DOF = 983,367			$N_x \times N_y \times N_z = 64 \times 64 \times 256$ DOF = 7,600,775		
		$N_n = 1$	$N_n = 2$	$N_n = 4$	$N_n = 1$	$N_n = 2$	$N_n = 4$
$p_n = 1$	$t_p = 32$	00:01:26	00:00:52	00:00:37	00:17:48	00:09:49	00:05:39
$p_n = 2$	$t_p = 16$	00:01:14	00:00:44	00:00:30	00:16:20	00:08:50	00:04:48
$p_n = 4$	$t_p = 8$	00:01:10	00:00:42	00:00:29	00:15:47	00:07:54	00:04:37
$p_n = 8$	$t_p = 4$	00:01:09	00:00:41	00:00:28	00:14:50	00:07:50	00:04:34
$p_n = 16$	$t_p = 2$	00:01:07	00:00:41	00:00:28	00:14:05	00:07:50	00:04:31
$p_n = 32$	$t_p = 1$	00:01:04	00:00:38	00:00:28	00:13:30	00:07:24	00:04:20

References

[1] L. T. Izu, J. R. H. Mauban, C. W. Balke, and W. G. Wier, Biophys. J. **80**, 88–102 (2001).
[2] L. T. Izu, W. G. Wier, and C. W. Balke, Biophys. J. **80**, 103–120 (2001).
[3] J. Schäfer, X. Huang, S. Kopecz, P. Birken, M. K. Gobbert, and A. Meister, Numer. Methods Partial Differential Equations **31**(1), 143–167 (2015).
[4] S. Kopez, A. Meister, B. E. Peercy, and M. K. Gobbert, Proc. Appl. Math. Mech. (PAMM) **18** (accepted (2018)).
[5] G. C. Kroiz, C. Barajas, M. K. Gobbert, and B. E. Peercy, Linkages of calcium induced calcium release in a cardiomyocyte simulated by a system of seven coupled partial differential equations, *Involve: A Journal of Mathematics*, submitted (2019).
[6] C. Barajas, M. K. Gobbert, G. C. Kroiz, and B. E. Peercy, Challenges and opportunities for the simulations of calcium waves on modern multi-core and many-core parallel computing platforms, *International Journal for Numerical Methods in Biomedical Engineering*, submitted (2018).

