

---

# Parallel Numerical Solution of Intracellular Calcium Dynamics

Chamakuri Nagaiah<sup>1</sup>, Sten Rüdiger<sup>2,3</sup>, Gerald Warnecke<sup>1</sup>, and Martin Falcke<sup>2</sup>

<sup>1</sup> Institute for Analysis and Numerics, Otto-von-Guericke University, 39106 Magdeburg, Germany. [Nagaiah.Chamakuri@student.Uni-Magdeburg.De](mailto:Nagaiah.Chamakuri@student.uni-magdeburg.de), [Gerald.Warnecke@Mathematik.Uni-Magdeburg.De](mailto:Gerald.Warnecke@mathematik.uni-magdeburg.de)

<sup>2</sup> Department of Theoretical Physics, Hahn-Meitner Institute, 14109 Berlin, Germany. [falcke@hmi.de](mailto:falcke@hmi.de)

<sup>3</sup> Institut für Physik, Humboldt-Universität zu Berlin, Newtonstr. 15, 12489 Berlin, Germany. [sten.ruediger@physik.hu-berlin.de](mailto:sten.ruediger@physik.hu-berlin.de)

**Summary.** We present a parallel numerical approach for intracellular calcium dynamics. Calcium is an important second messenger in cell communication. The dynamics of intracellular calcium is determined by the liberation and uptake by cellular stores as well as reactions with buffers. We develop models and numerical tools to study the liberation of calcium from the endoplasmic reticulum (ER). This process is characterized by the existence of multiple length scales. The modeling of the problem leads to a nonlinear reaction-diffusion system with natural boundary conditions in 2D. We used piecewise linear finite elements for the spatial discretization and time discretization by a linearly implicit Runge-Kutta scheme. We used the CHACO package for the domain decomposition. In our description the dynamics of IP<sub>3</sub>-controlled channels remains discrete and stochastic. It is implemented in the numerical simulation by a stochastic source term in the reaction diffusion equation. The strongly localized temporal behavior due to the on-off behavior of channels as well as their spatial localization is treated by an adaptive numerical method.

## 1 Introduction

Ca<sup>2+</sup> acts as an intracellular messenger regulating multiple cellular functions such as gene expression, secretion, muscle contraction or synaptic plasticity. The Ca<sup>2+</sup> signal employed by a variety of processes is a transient increase of the concentration in the cytosol. This is modeled by a system of reaction-diffusion equations with stochastic source terms for which we present numerical simulations.

In this article, we will outline the following important factors in the numerical solution of the problem: grid adaptivity, space and time discretization, coupling between space and time approximations, and parallelization. Briefly, it is very important to have a adaptive grid refinement in the area of clusters to obtain an efficient and fast numerical solutions. The finite element method is very suitable to handle these unstructured grids and complex geometry. We use a linearly implicit methods

to avoid nonlinear algebraic systems which arise for fully implicit methods after the time discretization. The classical embedding technique for ordinary differential equation integrators is employed to estimate the error in time. An automatic step size selection procedure ensures that the step size is as large as possible to guarantee the desired precision. To speed up the calculations parallelization is essential. Here the domain decomposition enters at the level of solution of algebraic system, see [1].

The paper is organized as follows. In the second Section we present the model which comprises calcium-buffer binding, diffusion and transport through the ER membrane. We will then introduce our method and strategies for grid adaptation, finite-element discretization and time-stepping in Section 3. Section 4 presents test results using sequential calculations and based on the domain decomposition method which is basic to our parallel code. Section 5 gives a short discussion of our work.

## 2 Governing Equations

The model consists of equations for the following deterministic quantities: calcium concentration in the cytosol and the ER as well as concentrations of several buffers. The current 2D model describes the concentrations of the involved chemical species in a thin layer on both sides of an idealized plane ER membrane. More details regarding the 2D modeling can be found in [2]. The evolution of concentrations will be determined by diffusion, transport of calcium through the ER membrane, and the binding and unbinding of buffer molecules to calcium:

$$\frac{\partial c}{\partial t} = D_c \Delta c + (P_l + P_c(r))(E - c) - P_p \frac{c^2}{K_d^2 + c^2} - \sum_i H_i(c, b_i), \quad (1)$$

$$\frac{\partial E}{\partial t} = D_E \Delta E + \gamma \left[ (P_l + P_c(r))(E - c) - P_p \frac{c^2}{K_d^2 + c^2} \right] - \sum_j K_j(c, b_{E,j}), \quad (2)$$

$$\frac{\partial b_i}{\partial t} = D_{b,i} \nabla^2 b_i + H_i(c, b_i), \quad i = s, m, d, \quad (3)$$

$$\frac{\partial b_{E,j}}{\partial t} = D_{E,j} \nabla^2 b_{E,j} + K_j(E, b_{E,j}), \quad j = s, m. \quad (4)$$

Here  $c$  is the concentration of  $\text{Ca}^{2+}$  in the cytosol,  $E$  is the concentration in the ER. The buffer concentration of bound calcium in the cytosol and the ER is given by  $b_i$  or  $b_{E,j}$ , respectively. We have  $i = s, d, m$  and  $j = s, m$ , where  $s$  denotes a stationary,  $d$  a dye and  $m$  a mobile buffers. All buffers are assumed to be distributed homogeneously in the initial state. Immobile buffers are modeled by setting their diffusion coefficient to zero. Total buffer concentrations in the cytosol and the ER are denoted by  $B_i$  or  $G_j$ , respectively. The buffer binding and unbinding of calcium is modeled by the usual mass-action kinetic terms:

$$H_i = k_{b,i}^+(B_i - b_i)c - k_{b,i}^- b_i, \quad K_j = k_{E,j}^+(G_j - b_{E,j})E - k_{E,j}^- b_{E,j}. \quad (5)$$

The second to fourth terms on the right hand sides of (1)-(2) model the transport of calcium through the membrane: leak current, current through  $\text{IP}_3$  controlled channels, and pump current, respectively. Channels are typically clustered on the membrane [2]. If a channel is open it contributes within the model to an effective circular source area given by the channel flux term

$$P_c(\mathbf{r}) = \begin{cases} P_{ch} & \text{if } \|\mathbf{r} - \mathbf{x}_n\| < R_n \text{ for a cluster } n, \\ 0 & \text{otherwise.} \end{cases}$$

Here the radius  $R_n$  of the cluster  $n$  with  $N_{open,n}$  open channels is then given by  $R_n = R_s \sqrt{N_{open,n}}$ . The parameter  $R_s$  is the source area of a cluster with one open channel. The position of the cluster is given by a fixed position  $\mathbf{x}_n$ .

An additional complexity of the model stems from the stochastic behavior of channel openings and closings, which needs to be incorporated into the computational approach. For an introduction to the hybrid algorithm to couple deterministic and stochastic simulations see the recent paper by [5].

### 3 Numerical Method

#### 3.1 Spatial Discretization Using Finite Elements

The domain  $\Omega \subseteq \mathbb{R}^2$  is a convex polygonal subset with piecewise smooth boundary  $\Gamma$ . The state variables  $c(\underline{x}, t)$ ,  $E(\underline{x}, t)$ ,  $b_m(\underline{x}, t)$  and  $b_{Em}(\underline{x}, t)$  are functions of space and time with values in  $\Omega \times [0, T]$ . We shall denote by  $L^2(\Omega)$  the space of square-integrable functions over  $\Omega$ . This space is equipped with the standard inner product  $\langle u, v \rangle = \int_{\Omega} uv \, dx$  and  $\|u\|_0 = \langle u, u \rangle^{1/2}$ . Next we define a Sobolev space of square integrable functions and derivatives

$$H^1(\Omega) = \{v \in L^2(\Omega), \partial_i v \in L^2(\Omega), 1 \leq i \leq d\}.$$

#### 3.2 Semi Discretization in Space

The partial differential equations can be written in the following general form

$$\begin{aligned} \frac{\partial \mathbf{u}(\mathbf{x}, t)}{\partial t} - \nabla \cdot (\mathbf{A}(\mathbf{x}) \nabla \mathbf{u}(\mathbf{x}, t)) + \mathbf{r}(\mathbf{u}(\mathbf{x}, t)) &= \mathbf{f} \text{ in } \Omega \times (0, T], \\ \mathbf{u}(\mathbf{x}, t) &= \mathbf{u}_0(\mathbf{x}) \text{ on } \Omega \times t = 0, \\ \mathbf{n} \cdot \nabla \mathbf{u}(\mathbf{x}, t) &= 0 \text{ on } \partial\Omega \times [0, T], \end{aligned} \tag{6}$$

where  $\mathbf{u}(\mathbf{x}, t)$  is unknown,  $\mathbf{A}(\mathbf{x}) > 0$  is the diffusion matrix and  $\mathbf{r}(\mathbf{u}(\mathbf{x}, t))$  is the reaction function. Letting  $V = H^1(\Omega)$ , multiplying the above equation for a given time  $t$  by  $v \in V$ , integrating over  $\Omega$  and using Green's formula, we get the *variational formulation*. Now let  $V_h$  be a finite dimensional subspace of  $V$  with basis  $\{w_1, \dots, w_N\}$ . Specifically we take continuous functions that are piecewise linear on a quasi-uniform triangulation of  $\Omega$  with mesh size  $h$ . Finally, we get a system of ordinary differential equations in the form

$$\mathbf{M} \dot{\mathbf{u}}_h + \mathbf{A} \mathbf{u}_h + \mathbf{s}(\mathbf{u}_h) = \mathbf{f}, \tag{7}$$

where  $\mathbf{M}$  is the mass matrix,  $\mathbf{A}$  is the stiffness matrix and  $\mathbf{s}(\mathbf{u}_h)$  is the vector depending on reaction term. The matrices are defined as follows

$$\mathbf{M} = \langle w_i, w_j \rangle, \quad \mathbf{A} = \langle \mathbf{A}(\mathbf{x}) \nabla w_i, \nabla w_j \rangle, \quad \mathbf{s}(\mathbf{u}_h) = \langle \mathbf{r}(\sum_{i=1}^N u_i(t) w_i(x)), w_j \rangle.$$

It is common practice to approximate the mass matrix  $\mathbf{M}$  by a diagonal matrix, which can be invertible easily. This is called a *lumping* process, see [4].

### 3.3 Temporal Time-stepping of Continuous Equations

The ordinary differential equation system, acquired from the semi discretization in space is solved numerically with finite difference methods. We considered the ODE problem

$$\frac{\partial \mathbf{u}}{\partial t} = \mathbf{G}(\mathbf{u}), \quad \mathbf{u}(t^0) = \mathbf{u}^0. \quad (8)$$

The notation for time step is  $\tau^i = t^{i+1} - t^i$  and  $\mathbf{u}^i$  to be the numerical solution at time  $t^i$ . The  $i$ -th time step of a W-method (linearly implicit Runge-Kutta type method) of order  $p$  with embedding of order  $\hat{p} \neq p$  has the form

$$(\mathbf{I} - \tau^i \gamma \mathbf{J}) \mathbf{k}_j = \mathbf{G} \left( t^i + \tau^i a_j, \mathbf{u}^i + \tau^i \sum_{l=1}^{j-1} b_{lj} \mathbf{k}_l \right) + \sum_{l=1}^{j-1} c_{lj} \mathbf{k}_l, \quad j = 1, \dots, s, \quad (9)$$

$$\mathbf{u}^{i+1} = \mathbf{u}^i + \tau^i \sum_{l=1}^s d_l \mathbf{k}_l, \quad \hat{\mathbf{u}}^{i+1} = \mathbf{u}^i + \tau^i \sum_{l=1}^s \hat{d}_l \mathbf{k}_l. \quad (10)$$

The method coefficients  $\gamma, a_j, b_{jk}, c_{jk}, d_j$ , and  $\hat{d}_j$  are chosen such that the local error of  $\mathbf{u}$  is of order  $\tau_i^{p+1}$ , the local error of  $\hat{\mathbf{u}}$  is of order  $\tau_i^{\hat{p}+1}$ , and these orders are independent of the matrix  $\mathbf{J}$  that is used. We assume  $p > \hat{p}$  which is reasonable since one would prefer to continue the integration with the higher order solution  $\mathbf{u}$ . In our computations we use a W-method with  $s = 3$  stages and for the coefficients, see [6].

After the  $i$ -th integration step the value  $\epsilon = \|\mathbf{u}^{i+1} - \hat{\mathbf{u}}^{i+1}\|$  is taken as an estimator of the local temporal error. A new time step  $\tau_{\text{new}}$  is computed by

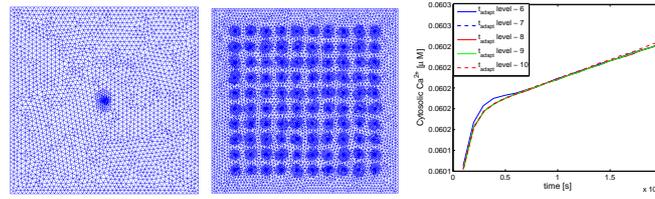
$$\bar{\tau} := \beta \tau^i \left( \frac{TOL_t}{\epsilon} \right)^{\frac{1}{\hat{p}+1}}, \quad \tau_{\text{new}} = \begin{cases} \beta_{\max} \tau^i, & \bar{\tau} > \beta_{\max} \tau^i, \\ \beta_{\min} \tau^i, & \bar{\tau} < \beta_{\min} \tau^i, \\ \bar{\tau}, & \text{otherwise.} \end{cases} \quad (11)$$

The parameter  $\beta > 0$  is safety factor. The factors  $\beta_{\min}$  and  $\beta_{\max}$  restrict time step jumps. If  $\epsilon < TOL_t$  we proceed to the next time step, otherwise the time step has to be shortened and repeated. Finally, after time discretization, we get system of algebraic equations in each stage. For solving the system in each stage we used the BiCGSTAB method with ILU preconditioning.

### 3.4 Grid Adaptivity

As spatial adaptivity criterion we used the  $Z^2$  error estimator of [8], see also [7]. For the refinement we used the following strategy. Let  $\lambda(T) \in \mathbb{N}_0$  be the refinement level of triangle  $T \in \mathcal{T}$ ,  $\lambda_{max} \in \mathbb{N}_0$  be a given maximum refinement level, and  $\phi_1, \dots, \phi_{\lambda_{max}}$  be given real numbers satisfying  $0 \leq \phi_1 \dots \leq \phi_{\lambda_{max}}$ . Here we used the scaled indicator  $\phi_T := \eta_{Z,T} / \sqrt{T}$ . For the initial grid and grid adaption we used the program package UG, [1]. We refine the mesh until minimum 4 grid points lie in the area of each channel. For the Test Cases 1 and 2 the initial triangulation a diameter of 700 nm for the triangle is considered.

**Test Case 1.** In this case we considered one cluster with 20 channels and the domain size is  $[0, 18000 \text{ nm}] \times [0, 18000 \text{ nm}]$ . The final mesh for this test case can be seen in the left hand Fig. 1.

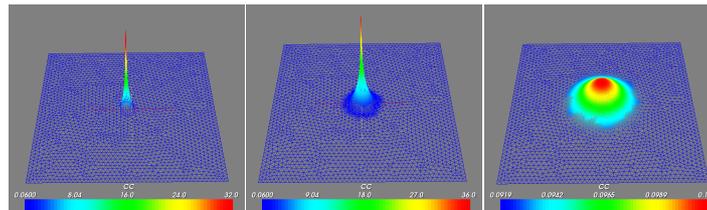


**Fig. 1.** Mesh level 6 for 1 cluster and 100 clusters, convergence result of cytosolic calcium at different adaptive levels.

**Test Case 2.** In this case we considered 100 clusters with a distance of  $4 \mu\text{m}$  and each cluster consists of 20 channels. The domain size is  $[0,48000 \text{ nm}] \times [0,48000 \text{ nm}]$ . The final mesh for this test case can be seen in the middle Fig. 1.

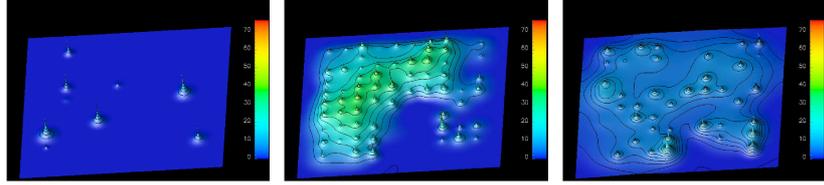
### 4 Numerical Results

In this subsection we will present the convergence results of one cluster with 1 opening channel. In all simulations we used the parameters  $D_c = 200 \mu\text{m}^2 \text{s}^{-1}$ ,  $D_E = 200 \mu\text{m}^2 \text{s}^{-1}$ ,  $D_m = 40 \mu\text{m}^2 \text{s}^{-1}$ ,  $D_s = 0.01 \mu\text{m}^2 \text{s}^{-1}$ ,  $P_{ch} = 3.0 \mu\text{m} \text{s}^{-1}$ ,  $P_l = 0.025 \mu\text{m} \text{s}^{-1}$ ,  $P_p = 200 \mu\text{m} \mu\text{M} \text{s}^{-1}$ ,  $R_s = 18 \text{ nm}$ ,  $K_d = 0.04 \mu\text{M}$  and initial solutions for  $c_0 = 0.06 \mu\text{M}$ ,  $E_c = 700 \mu\text{M}$ . First let us consider that in the numerical simulation one channel is open for a while. We tested the result with temporally adapted different grid levels. For different levels the average value of cytosol calcium concentration is shown in the right hand Fig. 1. The average value of the solution is calculated by using the integral average  $\bar{f} = \frac{1}{|\Omega|} \int_{\Omega} f(x) dx$ . In the next case, see Fig. 2, we have incorporated grid adaptivity during the intermediate time steps at mesh level 7. Here channel opening is considered in the stochastic regime. Initially mesh level 7 contains 2737 nodes and 5284 elements, at time  $t = 6.504 \text{ s}$  has 3216 nodes and 6242 elements, at time  $t = 8.92 \text{ s}$  it reaches to 18493 nodes and 36796 elements. In Fig. 3 the cytosolic calcium at different times with 100 clusters is shown.



**Fig. 2.** The numerical result of cytosolic calcium at time  $t = 6.504 \text{ s}$ ,  $6.68 \text{ s}$ ,  $8.92 \text{ s}$  in 1 cluster.

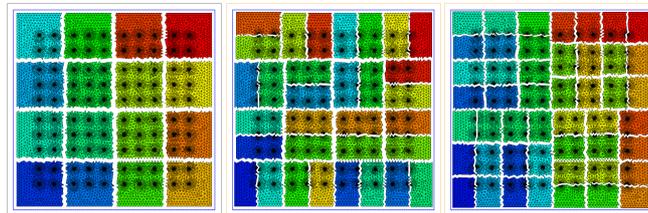
Here the channel opening is simulated stochastically.



**Fig. 3.** The numerical result of cytosolic calcium at times = 5.50 s, 6.13 s, 9.60 s in 100 cluster.

#### 4.1 Numerical Results Using Domain Decomposition Methods

In our numerical code to run the simulation time 100 s on a single processor, the CPU time takes around 50 days. To reduce the computational time and to be able to increase the number of mesh elements to millions the use of parallel computer architectures is mandatory. For the domain decomposition we used the graph partitioning package CHACO of [3]. The load balancing scheme Recursive Inertial Bisection (RIB) serves well for this problem. Load balancing has been achieved as follows: the meshes of level-0 to level-5 have been kept on one processor and the level-6 mesh has been distributed to all processors. The mesh decomposition to different processors is shown in Fig. 4.1. Computations for this problem have been carried out on HP-UX B.11.11 machines with 2GB RAM for each processor this is connected to a 64 node cluster with 3GFOLPS processor speed at our Institute.



**Fig. 4.** Domain decomposition using 16, 32 and 48 processors

Performance data of the simulations are presented in Table 1. The time step size is kept constant in all simulations for the sake of comparison. The first column shows the number of processors used and the last column shows the efficiency of the processors. This efficiency can be calculated using  $\frac{1}{P} \frac{T(1)}{T(P)}$ , where  $T(1)$  and  $T(P)$  are total CPU time for 1 processor and  $P$  processors. Efficiency is increased if we

**Table 1.** Comparison of CPU times using different processors

no. of procs	unknowns	time steps	cpu time	efficiency
1	133,296	10	26m 46s	-
16	133,296	10	2m 16s	0.7381
32	133,296	10	1m 2s	0.8095
48	133,296	10	38s	0.8805
56	133,296	10	32s	0.8962

increase the number of processors, because of the data structure of the programming package. The increase of the efficiency for 56 processors is 89.62%.

## 5 Conclusions

In this article we have presented sequential and parallel numerical results for intracellular calcium dynamics in 2 dimensions. In the sequential case, we presented the results of hybrid deterministic and stochastic models. In a test, we obtained good agreement between all mesh levels when channels open for a prescribed time. We observed that spatial adaptivity in time is important if channels open and close stochastically. It is challenging to extend the computations to higher numbers of clusters and 3 dimensions. Furthermore, we presented parallel numerical results using domain decomposition for a setup, where the channels open in a prescribed deterministic way. Here we obtained a reasonably accurate numerical solution upon increasing the number of processors. Extension of our parallel program to stochastic channel dynamics is in progress.

*Acknowledgement.* The authors gratefully acknowledge the Deutsche Forschungsgemeinschaft (DFG), Germany for financial support under the project grants (Wa 633/16-1, Fa 350/6-1).

## References

- [1] P. Bastian, K. Birken, S. Lang, K. Johannsen, N. Neuß, H. Rentz-Reichert, and C. Wieners. UG: A flexible software toolbox for solving partial differential equations. *Comput. Vis. Sci.*, 1:27–40, 1997.
- [2] M. Falcke. On the role of stochastic channel behavior in intracellular  $\text{Ca}^{2+}$  dynamics. *Biophys. J.*, 84(1):42–56, 2003.
- [3] B. Hendrickson and R. Leland. The CHACO user's guide 1.0. Technical Report SAND93–2339, Sandia National Laboratories, 1993.
- [4] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations*. Springer-Verlag, Berlin, 1994.
- [5] S. Rüdiger, J. W. Shuai, W. Huisinga, Ch. Nagaiah, G. Warnecke, I. Parker, and M. Falcke. Hybrid stochastic and deterministic simulations of calcium blips, 2006. In preparation.

- [6] B. A. Schmitt and R. Weiner. Matrix-free W-methods using a multiple Arnoldi iteration. *Appl. Numer. Math.*, 18:307–320, 1995.
- [7] R. Verfürth. *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*. Teubner and Wiley, Stuttgart, 1996.
- [8] O. C. Zienkiewicz and J. Z. Zhu. A simple error estimator and adaptive procedure for practical engineering analysis. *Internat. J. Numer. Methods Engrg.*, 24:337–357, 1987.