Higher order optimization and adaptive numerical solution for optimal control of monodomain equations in cardiac electrophysiology

Chamakuri Nagaiah[*] and Karl Kunisch[†]
Institute of Mathematics and Scientific Computing,
University of Graz, Heinrichstr. 36, Graz, A-8010, Austria.

March 25, 2010

Abstract

In this work adaptive and high resolution numerical discretization techniques are demonstrated for solving optimal control of the monodomain equations in cardiac electrophysiology. A monodomain model, which is a well established model for describing the wave propagation of the action potential in the cardiac tissue, will be employed for the numerical experiments. The optimal control problem is considered as a PDE constrained optimization problem. We present an optimal control formulation for the monodomain equations with an extra-cellular current as the control variable which must be determined in such a way that excitations of the transmembrane voltage are damped in an optimal manner.

The focus of this work is on the development and implementation of an efficient numerical technique to solve an optimal control problem related to a reaction-diffusions system arising in cardiac electrophysiology. Specifically a Newton-type method for the monodomain model is developed. The numerical treatment is enhanced by using a second order time stepping method and adaptive grid refinement techniques. The numerical results clearly show that super-linear convergence is achieved in practice.

Keywords: reaction-diffusion equations, monodomain model, PDE constraint optimization, adaptive FEM, Newton's optimization algorithm.

[*] `nagaiah.chamakuri@uni-graz.at`

[†] `karl.kunisch@uni-graz.at`

1 Introduction

Optimization problems governed by partial differential equations arise in many application areas of natural science and engineering. These problems, now called PDE constrained optimization problems, due to their problem size and/or complexity still present a significant challenge to efficient numerical realization. One such interesting application is the optimal control of the bidomain model equations in cardiac electrophysiology. The bidomain model [8, 18, 22] describes the electrical behavior of the cardiac tissue by a reaction-diffusion system coupled with an ordinary differential equations which model the ionic currents associated with the reaction terms. The equations model the fact that currents leaving the extracellular space by traversing the cellular membranes are the sources of the intracellular current density and vice versa. That is, currents leaving the intracellular space are acting as sources of the extracellular current density. The numerical solution of the bidomain equations is computationally expensive, see [25, 24], due to the high spatio-temporal resolution required to resolve the fast transients and steep gradients governing wavefront propagation in the heart. Assuming that the anisotropy ratios of the two spaces are equal leads to a reduced model, referred to as the monodomain model, which can be solved at a much lower computational expense by avoiding the time consuming solution of an elliptic PDE [17]. Under many circumstances of practical relevance the monodomain model can be considered to approximate the bidomain model fairly well [19, 15].

Cardiac arrhythmia refers to sudden, irregular patterns in the heart rhythm that may cause the heart to stop beating completely or slow down to the point where life is unsustainable. This must be treated immediately to avoid sudden cardiac death. The only way to reestablish a normal rhythm is to apply a strong electrical shock, a process called defibrillation. During defibrillation extracellular currents are injected via electrodes to establish an extracellular potential distribution which acts to reduce the complexity of the activity. This is achieved either by extinguishing all electrical activity, i.e. the entire tissue returns to its quiescent state, or gradients in V_m are smoothed out to drive the system to a less heterogeneous state which reduces the likelihood of triggering new wavefronts via "break" mechanisms when switching off the applied field. In the context of optimally control cardiac arrhythmias, it is essential to determine the control response to an applied electric field as well as the optimal extracellular current density that acts to damp gradients of transmembrane voltage in the system. The main objective of this work is the development and computer implementation of an efficient numerical technique to solve optimal control problems for monodomain equations.

The optimal control approach is based on minimizing a properly chosen cost functional $J(v, I_e)$ depending on the extracellular current I_e as input and on the transmembrane potential v as one of the state variables. First- and second-order derivatives of the reduced cost are derived using a Lagrangian based formalism.

Numerical technique for solving the optimal control problems requires combining a discretization technique with an optimization method. A traditional method to solve such problems is by using the optimize before discretize approach. In this case one expresses the continuous optimality system first before discretizing it. A second approach is to first discretize the differential equations and the cost and to solve the resulting nonlinear programming method with an efficient method. Clearly it is desirable to follow an approach were these two strategies commute or at least lead to very similar results. For the problem class under consideration here we carried out a detailed comparison for the 1D case in Nagaiah et.al [14]. In this present work we stick to the optimize before discretize technique.

In the numerical simulations there are many important factors which put a high demand on the computing time. These include different length and time scales of the reaction terms, strong nonlinearities caused by ionic currents, anisotropy related to the fiber orientation, and rapid changes of the potentials [7, 24] . We have chosen the finite element method for the spatial and higher order linearly implicit Runge-Kutta time stepping methods for the temporal discretization. The mesh has to be adapted, during the primal and dual solves, at each time step in all regions in order to preserve the accuracy of the solution. To control the spatial discretization error, a-posteriori error estimators are computed to steer the mesh improvement by refinement and coarsening in each time step during the primal and dual solves in optimization algorithm. The adaptivity strategy is based on the ZZ-error estimator [26] in our work. Various other forms of adaptive mesh refinement techniques were applied successfully for excitable media by varying the spatial or temporal resolution or both [5, 7, 12, 21]. At present our numerical experiments are based on allowing adaptivity in the spatial grids, while the time step is kept constant during the primal and dual solves. More details will be given in subsection 4.3. Our numerical realization is based on the public domain software package DUNE [1].

The article is organized as follows: in the next section the governing equations for the action potential and the behavior of the ionic current variables using ionic models are described. In section 3 the control problem is posed for the monodomain equations and the first and second order derivatives of the reduced cost are characterized. Also a brief description of a Newton's algorithm is given. The numerical approach for solving the primal and the adjoint state equations is presented in section 4. Numerical results for two test cases are discussed in section 5. Finally concluding remarks are given.

2 The governing equations

The monodomain model consists in a parabolic reaction-diffusion equation for the transmembrane potential v coupled with a system of ODEs for the gating variables. We set $Q = \Omega \times [0, T]$, where $\Omega \subset \mathbb{R}^d$, $d = 2$, denotes the cardiac tissue sample domain with Lipschitz boundary $\partial\Omega$. The system is given by

$$\begin{aligned} \frac{\partial v}{\partial t} &= \nabla \cdot \bar{\sigma_i} \nabla v - I_{ion}(v,w) + I_e(x,t) \quad \text{in } Q & (1) \\ \frac{\partial w}{\partial t} &= g(v,w) \quad \text{in } Q, & (2) \end{aligned}$$

where $v : Q \to \mathbb{R}$ is the transmembrane voltage, $w : Q \to \mathbb{R}^n$ represents the ionic current variables, $\bar{\sigma_i} : \Omega \to \mathbb{R}^{d \times d}$ is the intracellular conductivity tensors, $I_e(x,t)$ is an extracellular current density stimulus, $I_{ion}(v,w)$ is the current density flowing through the ionic channels and the function $g(v,w)$ determines the evolution of the gating variables.. The above mentioned Eq. (1) is a parabolic type equation and Eq. (2) is a set of ordinary differential equations which can be solved independently for each node.

In the absence of a conductive bath both intracellular and extracellular domains are electrically isolated along the tissue boundaries and homogeneous Neumann boundary conditions are appropriate to reflect this fact. The initial values of transmembrane voltage and ion current variables are prescribed by given constant values. Here the initial and boundary conditions are chosen as

$$\begin{aligned} \bar{\sigma_i} \nabla v \cdot \eta &= 0 \quad \text{on } \partial Q = \partial\Omega \times [0,T] & (3) \\ v(x,0) = v_0 \qquad & \text{and } \ w(x,0) = w_0 \quad \text{on } \ \Omega\,, & (4) \end{aligned}$$

where $v_0 : \Omega \to \mathbb{R}^d$ denotes the initial transmembrane potential and $w_0 : \Omega \to \mathbb{R}^d$ is the initial ionic current variables at time $t = 0$.

Ionic model

In our numerical experiments, we considered a phenomenological model, namely a variant of the Fitzhugh-Nagumo (FHN) model [20], which is constructed to reproduce the macroscopically observed behavior of the cells. The ionic activity is modeled by an ordinary differential equations. In this case, the $I_{ion}(v,w)$ term is a cubic polynomial function in terms of the transmembrane potential v and linear in terms of the gating variable w:

$$I_{ion}(v,w) = Gv(1 - \frac{v}{v_{th}})(1 - \frac{v}{v_p}) + \eta_1 v w\,, \tag{5}$$

$$g(v,w) = \eta_2(\frac{v}{v_p} - \eta_3 w)\,, \tag{6}$$

where $G, \eta_1, \eta_2, \eta_3$ are positive real coefficients, v_{th} is a threshold potential and v_p the peak potential.

3 Optimal control problem

In this section we specify the optimal control problem that is under consideration. We consider

$$\text{(P)} \qquad \begin{cases} \min J(v, I_e)\,, \\ \\ e(v, w, I_e) = 0 \quad \text{in} \;\; Q\,, \end{cases} \tag{7}$$

where v and w are the state variables, and I_e is the control variable of the optimal control problem. Here $Q = \Omega \times (0,T)$ denotes the space-time cylinder and the coupled systems of PDE and ODE constraints is expressed as $e(v, w, I_e) = 0$, where

$$e(v, w, I_e) = \begin{pmatrix} \nabla \cdot (\bar{\sigma}_i \nabla v) - \frac{\partial v}{\partial t} - I_{ion}(v, w) + I_e(x,t) \\ \\ \frac{\partial w}{\partial t} - g(v, w) \end{pmatrix}. \tag{8}$$

The initial conditions in (4) are explicitly enforced and the homogenous Neumann boundary condition in (3) will be realized in the variational setting of the FE discretization. We refrain here from entering a function space setting. This requires independent investigations which will be reported elsewhere.

The control variable I_e is chosen such that it is nontrivial only on the control domain Ω_{con} of Ω, i.e. $I_e : \Omega_{con} \times (0,T) \to \mathbb{R}$, and I_e is extended by zero on $(\Omega \setminus \Omega_{con}) \times (0,T)$. It will not be necessary to introduce extra notation for this purpose. It can be argued that for each $I_e \in L^2((0,T) \times \Omega_{con}; \mathbb{R})$ there exists a unique $(v, w) \in L^2(0,T; H^1(\Omega)) \cap C(0,T; L^2(\Omega)) \cap L^4(Q) \times L^2(0,T; H^1(\Omega)) \cap C(0,T; L^2(\Omega))$ with $u_t, w_t \in L^2(0,T; H^1(\Omega)^* + L^{\frac{4}{3}}(Q)$ such that $e(v(I_e), w(I_e), I_e) = 0$, see [4]. With $v(I_e)$ thus defined we introduce the reduced cost functional

$$\hat{J}(I_e) := J(v(I_e), I_e). \tag{9}$$

Next we turn to the choice of the cost functional. For this purpose we introduce the observation domain $\Omega_{obs} \subset \Omega$. The control objective consists in dampening out the excitation wave of the transmembrane voltage Ω_{obs}. We therefore set

$$J(v, I_e) \;=\; \frac{1}{2} \int_0^T \left(\int_{\Omega_{obs}} |v - z|^2 \, \mathrm{d}\Omega_{obs} + \alpha \int_{\Omega_{con}} |I_e|^2 \, \mathrm{d}\Omega_{con} \right) dt, \tag{10}$$

where α is the weight of the cost of the control, which is used to determine the influence of the corresponding components, Ω_{obs} is the observation domain. In this paper the numerical experiments are conducted with $z = 0$ which corresponds to the desire to dampen the wave in Ω_{obs}. The inclusion of the tracking type term z in the cost functional serves code-validation purposes, see Nagaiah et al. [13] for more numerical results.

To derive the first derivative of the reduced cost we use the Lagrange functional

$$\begin{aligned}\mathcal{L}(v,w,I_e,p,q) &= J(v,I_e)+\int_0^T\int_\Omega\left(\frac{\partial w}{\partial t}-g(v,w)\right)q\,\mathrm{d}\Omega\,\mathrm{d}t,\\ &+\int_0^T\int_\Omega\left(\nabla\cdot\bar{\sigma}_i\nabla v-\frac{\partial v}{\partial t}-I_{ion}(v,w)+I_e\right)p\,\mathrm{d}\Omega\,\mathrm{d}t.\end{aligned}$$

The first order optimality system is obtained by formally setting the partial derivatives of $\mathcal{L}$ equal to 0. We find

$$\mathcal{L}_v: \qquad v|_{\Omega_{obs}}+\nabla\cdot\bar{\sigma}_i\nabla p+p_t-(I_{ion})_v p-g_v q=0\,, \tag{11}$$

$$\mathcal{L}_w: \qquad -(I_{ion})_w p-q_t-g_w(v,w)q=0\,, \tag{12}$$

where the subscripts v and w denote the partial derivatives with respect to these variables. Further we obtain the

$$\text{terminal conditions: } p(T)=0 \quad q(T)=0, \tag{13}$$

$$\text{boundary conditions: } \bar{\sigma}_i\nabla p\cdot\eta=0 \quad \text{on } \partial Q\,, \tag{14}$$

$$\text{and the optimality condition: } \mathcal{L}_{I_e}: \quad \alpha I_e+p=0\,, \quad \text{on } \Omega_{con}\,. \tag{15}$$

The first order necessary optimality conditions consist of the coupled system of primal equations (1-2), adjoint equations (11-12), together with initial conditions (4), terminal conditions (13) and the optimality condition (15).

3.1 Newton's method

In this subsection, we present the inexact Newton method for solving the reduced optimization problem. In our case it is infeasible to set up the Hessian matrix. Rather we explain the necessary steps for the computation of "the action of the Hessian of the reduced cost" on a given vector. Once this is achieved approximate Hessian directions of appropriately discretized problems can be computed. This procedure was advocated e.g. in [9], Hinze et.al [10] for solving optimization problem governed by evolutionary partial differential equations.

We use the Lagrangian functional based approach to derive the second derivative of the reduced cost functional. Proceeding in formal terms it can be expressed as follows [10]:

$$\hat{J}''=\mathcal{L}_{I_eI_e}+\delta u^*\mathcal{L}_{uI_e}+\mathcal{L}_{I_eu}\delta u+\delta u^*\mathcal{L}_{uu}\delta u\,,$$

where $\delta u=-e_u^{-1}e_{I_e}$. In this case e_u^{-1} is only formal since it requires to invert a differential operator. Here, for brevity, we denote $u=(v,w)$ and $y=(u,I_e)$. We now introduce the matrix operator $T(I_e)$ and the second derivative of $\mathcal{L}$ as follows,

$$T(x)=\begin{pmatrix}-e_u^{-1}e_{I_e}\\ Id_{I_e}\end{pmatrix} \text{ and } \mathcal{L}_{yy}=\begin{pmatrix}\mathcal{L}_{uu} & \mathcal{L}_{uI_e}\\ \mathcal{L}_{I_eu} & \mathcal{L}_{I_eI_e}\end{pmatrix},$$

where Id_{I_e} is the identity operator in the control space. From these quantities the representation of the Hessian can be constructed by using the following formula

$$J^{''}(I_e) = T^{*}(I_e)\mathcal{L}_{yy}T(I_e), \tag{16}$$

where $T^{*}(I_e)$ is the adjoint operator to $T(I_e)$. Now we carry out these steps for the calculation of the second derivative of the reduced cost functional associated to the monodomain problem. In this process we have to evaluate the first derivatives, i.e. the sensitivities. We calculate these derivatives w.r.t state and control variables as follows:

$$\begin{aligned} e_u(\delta v, \delta w, \delta I_e) &= \begin{pmatrix} \nabla\cdot(\bar{\sigma}_i\nabla\delta v) - \delta v_t - [I_{ion}]_v\delta v - [I_{ion}]_w\delta w \\ \delta w_t - \frac{\eta_2}{v_p}\delta v + \eta_2\eta_3\delta w \end{pmatrix} & (17) \\ \text{and } e_{I_e}(\delta I_e) &= \begin{pmatrix} \delta I_e \\ 0 \end{pmatrix} & (18) \end{aligned}$$

where

$$\begin{aligned} [I_{ion}]_v &= \frac{G}{v_p v_{th}}\left[(v_{th}-v)(v_p-v) - v(v_p-v) - v(v_{th}-v)\right] + \eta_1 w\,, \\ [I_{ion}]_w &= \eta_1 v. \end{aligned}$$

The computation of the second derivative operator $\mathcal{L}$ applied to the vector $(\delta v, \delta w, \delta I_e)$ can be expressed as

$$\mathcal{L}_{yy}(\delta v, \delta w, \delta I_e) = \begin{pmatrix} \begin{pmatrix} \delta v|_{\Omega obs} - [I_{ion}]_{vv}p - \eta_1\delta w p \\ -\eta_1\delta v p \end{pmatrix} \\ \alpha\delta I \end{pmatrix} \tag{19}$$

$$\text{where } [I_{ion}]_{vv} = \frac{2G}{v_p v_{th}}\delta v\left[v - (v_p - v) - (v_{th} - v)\right].$$

We note that in order to obtain the action of the Hessian on a given vector, one linearized primal problem and one linearized adjoint equation have to be solved. The basic steps to compute the action of the Hessian are summarized next.

1. Compute the first derivative $\hat{J}^{'}(u^n) = \alpha I_e(x,t) + p$, which requires one primal and one dual solve.

2. In each step iteratively evaluate the action of $\hat{J}^{''}(u^n)$ on δ^n which is done using the following sequence of computations.

   (a) solve the linearized primal equation for $\delta v, \delta w$ using δI_e^k

   $$\begin{pmatrix} \nabla\cdot(\bar{\sigma}_i\nabla\delta v) - (\delta v_t + [I_{ion}]_v\,\delta v + [I_{ion}]_w\,\delta w) \\ \delta w_t - \frac{\eta_2}{v_p}\delta v + \eta_2\eta_3\delta w \end{pmatrix} = \begin{pmatrix} -\delta I_e^k \\ 0 \end{pmatrix}$$

(b) evaluate $(z_1, z_2) := \mathcal{L}_{yy}(\delta v, \delta w, \delta I_e)$ from eq. (19)

(c) solve the adjoint equation with (z_1, z_2) as r.h.s i.e.

$$\begin{pmatrix} \nabla \cdot \bar{\sigma}_i \nabla w_1 + w_{1_t} - [I_{ion}]_v w_1 - \frac{\eta_2}{v_p} w_2 \\ -[I_{ion}]_w w_1 - w_{2_t} + \eta_2 \eta_3 w_2 \end{pmatrix} = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}$$

(d) finally compute the action of $\hat{J}''(u^k)$ on δI_e^k i.e $-w_1 + \alpha \delta I_e$

In this way one can compute the action of the Hessian of the reduced cost. Consider the system of step (7) in Algorithm 1, given in the Appendix. Its dimension is that of the control space dimension. To evaluate this step one must use an iterative method, e.g. a linear conjugate gradient method.

4 Numerical discretization

For the approximation of the optimality system (1-2) and (11-12) we use a finite element method for the spatial discretization. This results in an initial value problem for a system of ordinary differential equations. The time discretization is based on an explicit Euler method for the ODE equations and a linearly implicit Runge-Kutta method for the parabolic equations.

4.1 Space discretization using FEM

In this subsection we give a brief description of the piecewise linear finite element discretization to solve the monodomain equations. We commence with the primal problem in variational form: find $v : [0, T] \to H^1(\Omega)$ and $w : [0, T] \to L^2(\Omega)$ such that for a.e. $t \in (0, T)$

$$\langle \nabla \cdot \bar{\sigma}_i \nabla v, \varphi \rangle = \left\langle \left(\frac{\partial v}{\partial t} + \int_Q I_{ion}(v, w) - I_e \right), \varphi \right\rangle \text{ for all } \varphi \in H^1(\Omega) \qquad (20)$$

$$\left\langle \frac{\partial w}{\partial t}, \varphi \right\rangle = \langle g(v, w), \varphi \rangle .$$

Let $V_h \subset H^1(\Omega)$ be the finite dimensional subspace of piecewise linear basis functions with respect to the spatial grid.

The approximate solution $\mathbf{v}$ is expressed in the form $\mathbf{v(t)} = \sum_{\mathbf{i=0}}^{\mathbf{N}} \mathbf{v}_{\mathbf{i}} \omega_{\mathbf{i}}$ where $\{\omega_i\}_{i=1}^N$ denote the basis functions. This results in the following system of nonlinear ordinary differential equations:

$$
\begin{aligned}
\mathbf{M}\frac{\partial \mathbf{v}}{\partial t} &= -\mathbf{A_i}\mathbf{v} - \mathbf{I_{ion}}(\mathbf{v},\mathbf{w}) + \mathbf{I_e}\,, && (21)\\
\frac{\partial \mathbf{w}}{\partial t} &= \mathbf{g}(\mathbf{v},\mathbf{w})\,, && (22)\\
\mathbf{v}(0) = \mathbf{0}\,, &\quad \mathbf{w}(0) = \mathbf{0}\,, && (23)
\end{aligned}
$$

where $\mathbf{A_i}$ is the stiffness and $\mathbf{M}$ is the mass matrix, and $\mathbf{I_{ion}}$ and $\mathbf{I_e}$ are vectors defined by $\mathbf{I_{ion}} = \{\langle I_{ion}, \omega_j\rangle\}_{j=1}^{N}$ and $\mathbf{I_e} = \{\langle I_e, \omega_j\rangle\}_{j=1}^{N}$, respectively.

Space discretization of dual problem

We use an analogous derivation as for the primal problem and obtain the following semi-discrete form of the dual equations:

$$
\begin{aligned}
\mathbf{M}\frac{\partial \mathbf{p}}{\partial t} &= -\mathbf{M}_{obs}\mathbf{v} + \mathbf{A_i}\mathbf{p} + \mathbf{M}(\mathbf{I_{ion}})_{\mathbf{v}}\mathbf{p} + \mathbf{M}\mathbf{q}g_v\,, && (24)\\
\frac{\partial \mathbf{q}}{\partial t} &= -g_w^T(v,w)\mathbf{q} - (I_{ion})_w\mathbf{p}\,, && (25)\\
\mathbf{p}(T) = \mathbf{0}\,, &\quad \mathbf{q}(T) = \mathbf{0}\,, && (26)
\end{aligned}
$$

where $\mathbf{M}_{obs}$ is a mass matrix in observation domain.

To solve the semi-discretized primal and the dual problems (21-23) and (24-26), we first approximate the ODE system solution at the current time step. This gives the ionic current variable update w while solving the primal problem, which is subsequently used to update the equation (21). Analogously, the adjoint varible update q is used in the equation (24). In our numerical computations the primal problem is solved by decoupling the system as follows:

$$
\begin{aligned}
\text{step-1 :}\quad & \mathbf{w^{n+1}} = \mathbf{w^n} + \Delta t\, \mathbf{g}(\mathbf{v^n},\mathbf{w^n})\,, && (27)\\
\text{step-2 :}\quad & \mathbf{M}\frac{\partial \mathbf{v^n}}{\partial t} = -\mathbf{A_i}\mathbf{v^n} - (\mathbf{I_{ion}}(\mathbf{v^n},\mathbf{w^{n+1}}) - \mathbf{I_e})\,. && (28)
\end{aligned}
$$

Analogously, the dual problem is decoupled as follows:

$$
\begin{aligned}
\text{step-1 :}\quad & \mathbf{q^n} = (1-\Delta t\,\eta_2\eta_3)\mathbf{q^{n+1}} + \Delta t\,\eta_1\mathbf{v^{n+1}}\mathbf{p^{n+1}}\,, && (29)\\
\text{step-2 :}\quad & \mathbf{M}\frac{\partial \mathbf{p^n}}{\partial t} = -\left(\mathbf{M}_{obs}\mathbf{v^{n+1}} - \mathbf{A_i}\mathbf{p^n}\right) + \mathbf{M}(\mathbf{I_{ion}})_{\mathbf{v}}\mathbf{p^n} + g_v\mathbf{M}\mathbf{q^{n+1}}\,. && (30)
\end{aligned}
$$

4.2 Time discretization using linearly implicit RK methods

In this subsection we give a brief description of the time discretization for solving the systems of ordinary differential equations which we now express in the following form:

$$\mathbf{M}\frac{\partial \mathbf{u}}{\partial t} = \mathbf{F}(\mathbf{u}), \qquad \mathbf{u}(t^0) = \mathbf{u^0}. \tag{31}$$

To solve (31), we introduce discrete steps in the time interval $[0, T]$:

$$0 = t^0, t^1, \dots, t^n = T\,,$$

which are not necessarily equidistant. We further set $\tau^i = t^{i+1} - t^i$ and denote by $\mathbf{u}^i$ the numerical solution at time t^i. For time discretization linearly implicit Runge-Kutta methods, specifically Rosenbrock methods, are used. These belong to a large class of methods which try to avoid the nonlinear system and replace it by a sequence of linear ones. For the construction of the Jacobian matrix we used exact derivatives of the vector $\mathbf{F}(\mathbf{u})$. For our computations the *ROS2* method was employed which has two internal stages to solve in each iteration see [11]. After the time discretization one ends up with a system of linear equations. For solving this linear system the BiCGSTAB method with ILU preconditioning is used.

4.3 Spatial grid adaptivity

The adaptive mesh refinement(AMR) algorithm uses a hierarchy of properly nested levels. It automatically refines or coarsens the mesh to achieve a solution having a specified accuracy in an optimal fashion. Here we used the AMR technique based on the Z^2 error indicator of Zienkiewicz and Zhu [26]. See also [23] for a more detailed description of error estimators. The full spatial and temporal discretization leads to an approximate solution v_h^t with $v_h^t(\cdot, t_i) \in V_h$ at the discrete time points t_i, $i = 0, \dots, M$ where the time integration scheme is evaluated. To recall the Z^2 error indicator, let v_i be the exact solution of the given problem.

??? in v_i what is the i, v^t????

We denote by W_h the space of all piecewise linear vector-fields and set $X_h := W_h \cap C(\Omega, \mathbb{R}^2)$. Denote by v and v_h the unique solution of problems (1) and (20), respectively. In this case $\|Gv_h - \nabla v_h\|_{L^2(T)}$ can be used as an error estimator, where Gv_h is an easily computable approximation of ∇v_h.

Let $Gv_h \in X_h$ be the $\langle \cdot, \cdot \rangle_h$-projection of ∇v_h onto X_h. It can computed by a local averaging of $\nabla v_{h|T}(\mathrm{x}_i)$ as follows

$$Gv_h(\mathrm{x}_i) = \sum_{T \subset D_x} \frac{|T|}{|D_x|} \nabla v_{h|T}(\mathrm{x}_i)\,. \tag{32}$$

Here, D_x is the union of the triangles having x as a vertex. We finally set the error indicator locally and globally respectively as follows

$$\eta_{Z,T} := \|Gv_h - \nabla v_h\|_{L^2(T)}\;, \tag{33}$$

and

$$\eta_Z := \left\{ \sum_{T \in \mathcal{T}_h} \eta_{Z,T}^2 \right\}^{1/2} . \tag{34}$$

The Z^2 indicator $\eta_{Z,T}$ is an estimate for $\|\nabla v_h^t(\cdot, t_i) - \nabla v^t(\cdot, t_i)\|_{L^2(T)}$, see Verfürth [23] for complete details. 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 $\phi_1, \ldots, \phi_{\lambda_{max}}$ be given real numbers satisfying $0 \leq \phi_1 \ldots \leq \phi_{\lambda_{max}}$. We set $\phi_0 = 0$ and $\phi_{\lambda_{max}} = \infty$. With the choice of $\phi_1, \ldots, \phi_{\lambda_{max}}$ one controls the structure of the grid. If we set $\phi_1 = \ldots = \phi_{\lambda_{max}} = 0$ this leads to a uniform triangulation of level λ_{max}. We set $\phi_1 = \ldots = \phi_{\lambda_{max}} = 10^{-2}$ in our numerical computations. Here we used the scaled indicator

$$\phi_T := \eta_{Z,T} / \sqrt{|T|} \, . \tag{35}$$

Suppose that an initial coarse triangular grid is constructed using a grid generator. A triangle T is marked for

1. refinement if $\phi_T > \phi_{\lambda(T)}$ and $\lambda(T) < i$ for $i = 0, \ldots, \lambda_{max}$,
2. coarsening if $\phi_T < \frac{\phi_{\lambda(T)}}{100}$ and $\lambda(T) > i$ for $i = 0, \ldots, \lambda_{max}$,

where ϕ_T is calculated according to Eq. (35).

So far we explained adaptivity for the direct problem. Let us turn to the optimal control problem next. One prominent approach in the literature is the dual weighted residual methodology as presented in [3] as a technique to derive a posteriori error estimates for both time dependent and stationary problems governed by partial differential equations. Another approach uses grid adaptivity based on a posteriori error estimate of the cost functional [2, 6]. In our computational work we proceeded as follows. We used AMR in space only and assume that, during the optimization iterations, the control variable lives always on the coarse grid dimension where we start the simulations. This can be related to the fact that in practice in is difficult to manipulate the control mechanism spatially. The initial coarse grid, denoted by $\mathbf{L}_0$ is constructed by a grid generator. Finer levels $\mathbf{L}_i$ for $i > 0$ are constructed recursively from the coarser levels L_{i-1}. The tolerance for spatial grid refinement is set $tol_x = 10^{-2}$. The Z^2 error estimator is called for every 3 time intervals during the primal solve. It adjusts the spatial grid by refining and coarsening, depending on the estimated spatial solution error of the elements. Accordingly, the control variable is interpolated based on the new grid construction. We used the same strategy to solve to adjoint problem with the spatial grid being adjusted according to the adjoint solution. In the Newton optimization algorithm, to evaluate the matrix-vector product in each iteration of the inner loop, see algorithm (1), the coarse grid dimension is used to up-date the control solution. Thus our Newton direction is an inexact one: it is determined from solving the Newton system on the coarse grid iteratively up to a certain tolerance tol_{New}. We set $tol_{New} = 10^{-3}|\hat{J}'(I_{e,k})|$. Since $|\hat{J}'(I_{e,k})| \to 0$ in case of convergence to a critical point we can expect to obtain superlinear convergence of the overall numerical optimization strategy. Of course, this can be achieved only if function and gradient evaluations are sufficiently accurate.

For our numerical experiments, we developed an optimization code based on the public domain software package DUNE [1].

5 Numerical results

In this section we present numerical results based on the adaptive and second order methods. We shall see that the proposed methodology is capable of dampening an excitation wave of the transmembrane potential by properly applying an extracellular potential, even if the control domain is relatively small. The numerical experiments were carried out using different static grid sizes and as well as the AMR technique. Here we consider 4 msec of simulation time. The computational domain $\Omega = [0, 1] \times [0, 1]$ and various relevant subdomains are depicted in Figure 1. The excitation domain and control domains are Ω_{exi} and $\Omega_{con} = \Omega_{con1} \cup \Omega_{con2}$. Further Ω_{f1}, Ω_{f2} are neighborhoods of $\Omega_{con1}, \Omega_{con2}$ and the observation domain is $\Omega_{obs} = \Omega \backslash (\Omega_{f1} \cup \Omega_{f2})$.

In the computations equidistant time steps are used. For the spatial adaptive grid refinement, a uniform 60×60 triangular grid is considered as coarse grid and the subsequent multi level grids are constructed based on the primal or adjoint solution. In simulations the weight of the cost of the control is fixed to be $\alpha = 1 \cdot 10^{-3}$ and the iterations were terminated when $\|\nabla J_k\|_\infty \leq 10^{-3}(1 + |J_k|)$, is satisfied or the difference between two successive optimization iterations of cost functional minimization value is less than 10^{-3}. If this condition is not met within the prescribed iterations, let us say 100 optimization iterations, then we terminate the simulations. The line search algorithm starts with initial step length $\alpha = 1$ for the inexact Newton method and is reduced by a factor of 2 for subsequent rejected step sizes, see Nocedal and Wright [16]. For the computational set up, we considered the domains $\Omega_{exi} = [0.498, 0.502] \times [0.498, 0.504]$, $\Omega_{con1} = [0.3, 0.4] \times [0.45, 0.55]$, $\Omega_{con2} = [0.6, 0.7] \times [0.45, 0.55]$, $\Omega_{f1} = [0.28, 0.42] \times [0.43, 0.57]$ and $\Omega_{f2} = [0.58, 0.72] \times [0.43, 0.57]$.

Parameters used in the simulation

The following parameter values [7] are used for our numerical experiments: $\sigma_{il} = 3{\cdot}10^{-3}\,\Omega^{-1}cm^{-1}$, $\sigma_{it} = 3.1525{\cdot}10^{-4}\,\Omega^{-1}cm^{-1}$ $G = 1.5\,mS/cm^2$, $v_{th} = 13\,mV$, $v_p = 100\ mV$, $\eta_1 = 4.4\ mS/cm^2$, $\eta_2 = 0.012$, $\eta_3 = 1$.

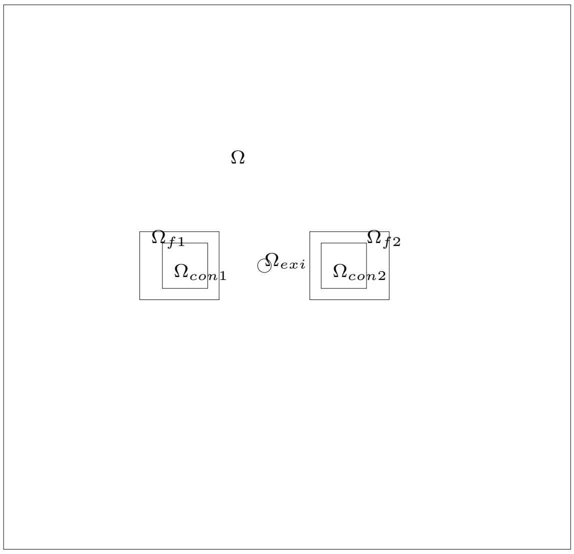

Figure 1: Control and excitation region at the cardiac domain

Test case 1

First we discuss the results based on different static grid sizes and the AMR grid. We use a 60×60 triangular grid, consisting of 13,924 elements and 7,081 nodal points, an 80×80 triangular grid, consisting of 24,964 elements and 12,641 nodal points, a 100×100 triangular grid, consisting of 40,000 elements and 20,201 nodal points, and AMR coarse grid which is a 60×60 triangular grid.

The corresponding L^2 norms of the gradients of the cost functional and the minimum values of the cost functional themselves are depicted in Figure 2. A logarithmic scale is considered for the presentation of L^2 norms of the gradients of the cost functional . As expected for a Newton method, the optimal solutions are obtained within a rather small number of iterations. For each one of the respective grids convergence was obtained within 8 or 9 iterations. The values of the two additive terms of the cost, $\frac{1}{2}\int \|V_m\|^2 \, \mathrm{d}x \, \mathrm{d}t$ and $\frac{\alpha}{2}\int \|I_e\|^2 \, \mathrm{d}x \, \mathrm{d}t$, are depicted in Figure 3. Note that the two terms are of approximately the same order of magnitude. We observed that the control action has a strong impact at the beginning of iterations where the excitation of the wave front is dampened out and decreases with time, see Figure 6.

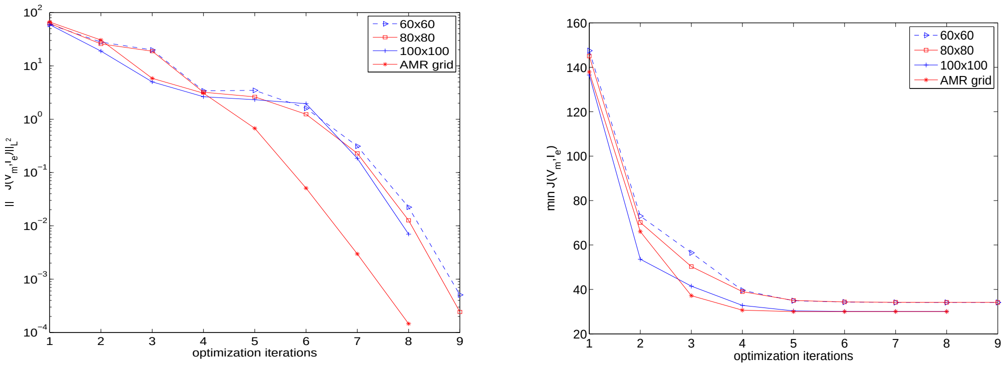

Figure 2: The norm of the gradient and minimum value of the cost functional are shown on left and right respectively for $T = 4$ msec of simulation time.

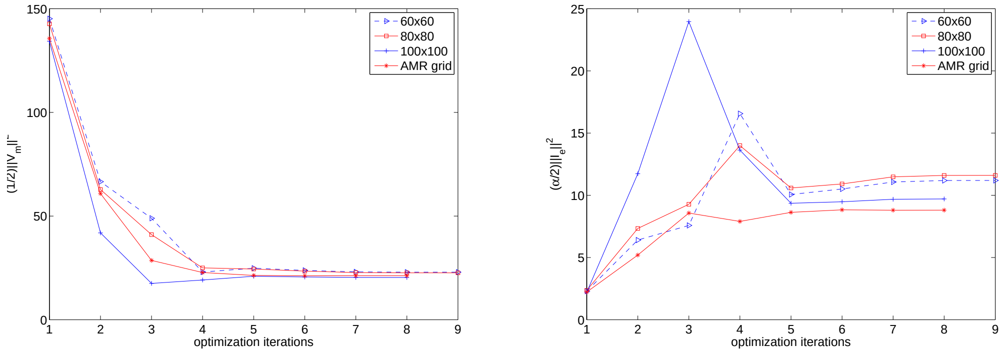

Figure 3: The minimum value of $\int |V_m|^2 \mathrm{d}x\mathrm{d}t$ and $\int |I_e|^2 \mathrm{d}x\mathrm{d}t$ are on left and right respectively for $T = 4$ msec of simulation time.

grids	CPU time	primal solves(s)	dual solves(s)	opt. solves(s)	avg inner CG iters
60×60	58min 51sec	93.64	87.24	3264.63	15.7
80×80	2h 2min 54sec	197.03	180.46	6813.89	17.7
100×100	3h 13min 14sec	323.89	292.27	10759.4	20
AMR grid	1h 13min 19sec	598.52	551.26	3187.07	17

Table 1: Computational time and an average inner CG iterations in Newton's algorithm are presented for all grids.

The total CPU time for solving the optimization problem, as well as the CPU time for all primal solves, all dual solves, and for all setup and evaluation times of the Hessian steps is given in the first 5 columns of Table 1. The last column contains the average number of CG-steps required for obtaining the inexact Newton algorithm. We note that the computational time for the gradient of the cost, which requires one primal and one dual solve per iteration, is much cheaper than updating the control variable on the basis of second order information. For all grids, updating the control variable on the basis of the second order system, needs approximately 18 times more CPU time than computing the first derivative. This is the expensive step in our calculations.

However, as we experienced in our earlier work, minimization on the basis of only first order information requires at least several hundred iterations before comparable accuracy can be achieved. Let us also mention that additional Newton steps, beyond those which are documented, allow to further decrease ∇J which we could not observe with the nonlinear CG algorithm [13].

The AMR technique takes 598 seconds to solve the primal problem, out of which it needs 96.87 seconds for grid adaption. For the dual solve the grid adaption process takes 81.96 seconds. We can observe that AMR takes less overall CPU time and achieves results comparable to the 100 × 100 grid. Obviously the AMR technique shows a good improvement over the static grids for the current problem.

60 × 60		80 × 80		100 × 100		AMR grid	
$\|\|\nabla J\|\|$	$\frac{\|\|\nabla J\|\|_{i+1}}{\|\|\nabla J\|\|_i}$	$\|\|\nabla J\|\|$	$\frac{\|\|\nabla J\|\|_{i+1}}{\|\|\nabla J\|\|_i}$	$\|\|\nabla J\|\|$	$\frac{\|\|\nabla J\|\|_{i+1}}{\|\|\nabla J\|\|_i}$	$\|\|\nabla J\|\|$	$\frac{\|\|\nabla J\|\|_{i+1}}{\|\|\nabla J\|\|_i}$
59.9128	–	62.5567	–	59.3120	–	66.5305	–
27.9293	0.4661	25.8370	0.4130	18.9710	0.3199	30.5083	0.4586
19.8697	0.7114	18.8618	0.7300	5.0035	0.2637	5.8358	0.1913
3.40178	0.1712	3.20784	0.1700	2.64203	0.5280	3.07498	0.5269
3.47448	1.0213	2.62626	0.8187	2.31805	0.8774	0.67403	0.2192
1.60263	0.4612	1.23709	0.4710	1.95699	0.8442	0.05105	0.0758
0.31174	0.1945	0.22913	0.1852	0.18502	0.0945	0.00180	0.0353
0.02236	0.0717	0.01267	0.0553	0.00700	0.0378	0.00010	0.0556
0.00050	0.0226	0.00024	0.0191	–	–	–	–

Table 2: Optimization iterations, norm of gradient of cost functional and order of convergence for the different grid constructions are presented using inexact Newton's algorithm.

Table 2 confirms superlinear convergence of the inexact Newton method for all grid constructions at the end of iterations. The line search algorithm takes small step lengths during the initial iterations and full step lengths after 3 or 4 iterations which leads to superlinear convergence.

The 2D contour snap shots of the uncontrolled solution, and of the optimally controlled solutions are shown in Figures 4, and 5 at times 0.3 msec, 2 msec and 4 msec for the 100 × 100 mesh. We can see that the uncontrolled wave front of the transmembrane voltage spreads almost over the entire domain during the time interval from 0 msec to 4 msec, see in Figure 4. The corresponding controlled action potential is first slowly dampened at 0.3 msec and is almost completely

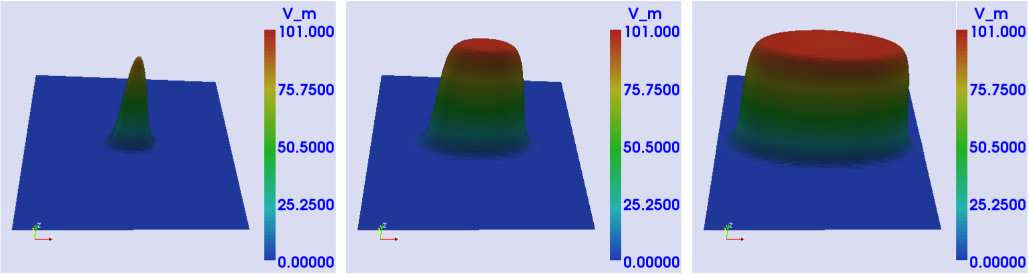

Figure 4: Uncontrolled solution (V_m) at 0.3 msec, 2 msec and 4 msec of simulation time respectively.

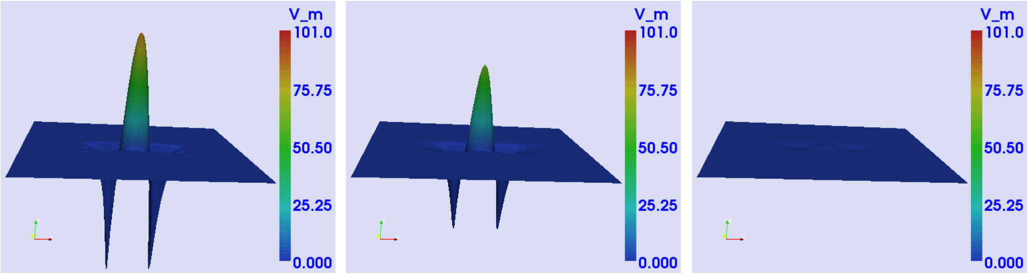

Figure 5: Controlled solution ($(V_m)_{opt}$) at 0.3 msec, 2 msec and 4 msec of simulation time respectively.

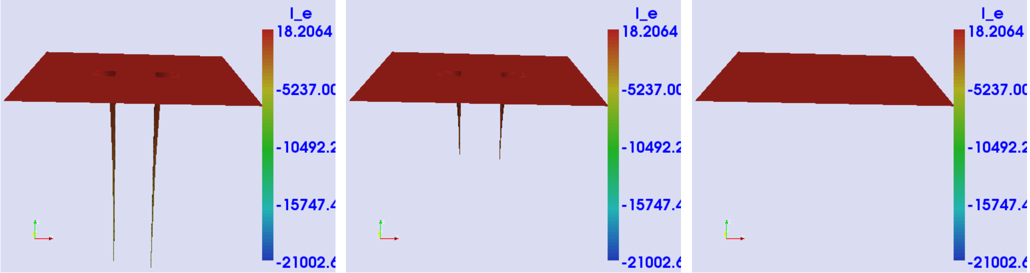

Figure 6: Controlled (I_e) at 0.3 msec, 2 msec and 4 msec of simulation time respectively.

damped at 4 msec. The control action has a tremendous effect on the action potential at the beginning to control and dampens the excitation wave It has much less effect at 4 msec time, see Figure 6. The AMR grid of the uncontrolled solution can be seen in Figure 7. The refinement/coarsening strategy follows the interface of the wave propagation. For the first optimization iteration the fine grid for the primal solve consists of 34,038 elements and 17,138 number of nodal points. At the last optimization iteration the fine grid consists of 21,700 number of elements and 10,969 number of nodes where the maximum level of refinement is 5. The AMR grids corresponding to the optimal state solution are shown in Figure 8. We can note that the grid at time $t = 4$ msec is equivalent to the coarse grid, which

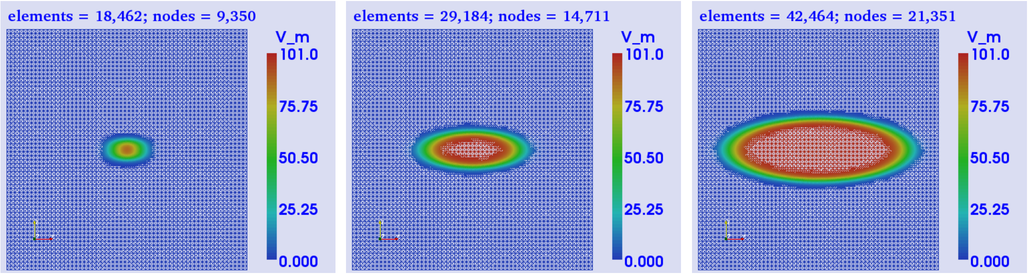

Figure 7: Uncontrolled solution (V_m) grid view at 0.3 msec, 2 msec and 4 msec of simulation time respectively for AMR technique.

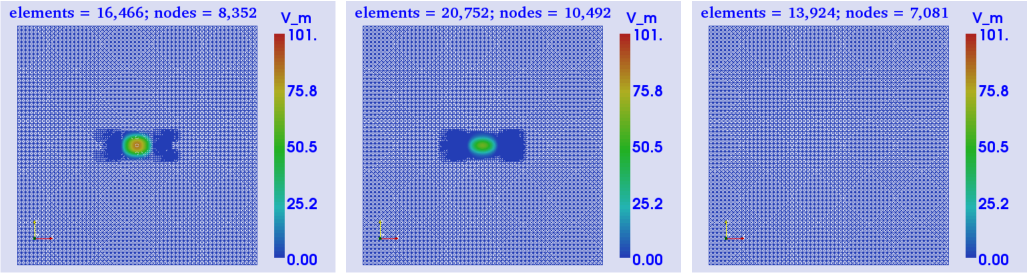

Figure 8: Controlled solution (V_m) grid view at 0.3 msec, 2 msec and 4 msec of simulation time respectively for AMR technique.

we considered as the level-0 grid. This is consistent with the fact that there is no further wave propagation within the domain. Finally, the presented numerical results evidently show that the excitation wave propagation is dampened out by properly adjusting the extracellular potential using the optimization algorithm. Further, the AMR method shows a good computational improvement over the static grids.

Test case 2

In this test case the placement of control domain is changed and isotropic conductivity tensors are considered. With respect to the control domains the observation domains are chosen in an asymmetric manner. More specifically, $\Omega_{con1} = [0.3, 0.4] \times [0.45, 0.55]$, $\Omega_{con2} = [0.5, 0.6] \times [0.53, 0.6]$, $\Omega_{obs1} = [0.25, 0.45] \times [0.40, 0.60]$ and $\Omega_{obs2} = [0.45, 0.65] \times [0.47, 0.65]$. In this test case the weight of the cost of the control is reduced to $\alpha = 5 \cdot 10^{-4}$ to achieve the dampening of exciting wave propagation.

The L^2 norms of the gradient and the minimum value of the cost functional are depicted in Figure 9 for different grid constructions. For all static grids 7 optimization iterations are required, while the AMR method takes 6 iterations. In this case the AMR method shows a clear improvement over the static grids in terms of fast converging to the optimal solution. The corresponding additive values of $\frac{1}{2}\int \|V_m\|^2\,\mathrm{d}x\mathrm{d}t$ and $\frac{\alpha}{2}\int \|I_e\|^2\,\mathrm{d}x\mathrm{d}t$ are depicted in Figure 10.

We observed that the control term has an especially strong impact during the first minimization iterations, during which the excitation of the wave front is dampened out. This can be seen on the right hand side of Figure 10.

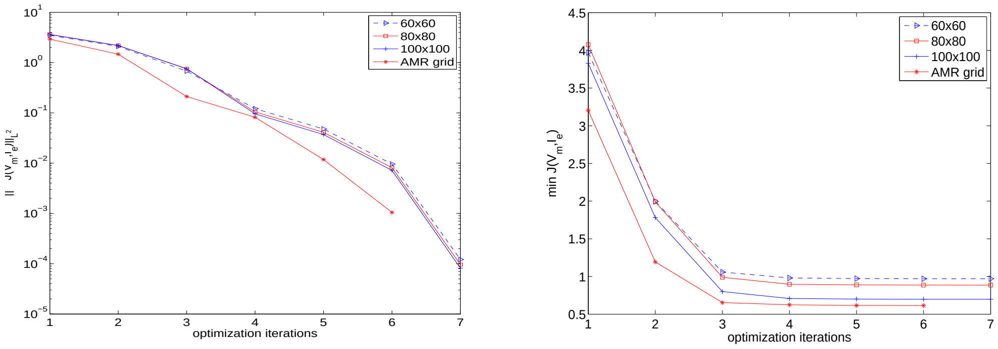

Figure 9: The norm of the gradient and minimum value of the cost functional are shown on left and right respectively for $T = 4$ msec of simulation time.

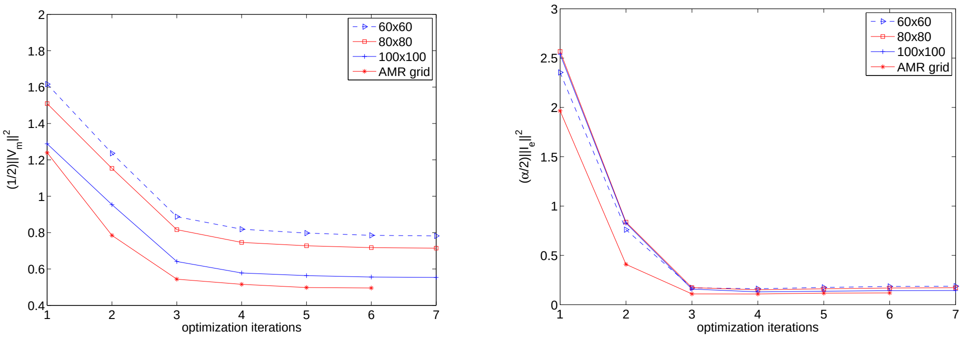

Figure 10: The minimum value of $\int |V_m|^2 \mathrm{d}x\mathrm{d}t$ and $\int |I_e|^2 \mathrm{d}x\mathrm{d}t$ are on left and right respectively for $T = 4$ msec of simulation time.

2D contour snap shots of the uncontrolled solution and the optimally controlled solution are shown in Figures 11, and 12 at times 0.3 msec, 2 msec and 4 msec the 100 × 100 mesh. We can see that the uncontrolled wave front of the transmembrane voltage spreads uniformly in the x- and y-directions over the domain during the time interval from 0 msec to 4 msec, see in Figure 11. In contrast to the Test case (1), we can observe that the controlled action potential is dampened much more rapidly, which is due to the isotropic conductivity tensors, $\sigma_{il} = \sigma_{it} = 3 \cdot 10^{-3}$ and the fact that the control cost parameter is reduced by a factor of 2. Also observe that the control domain that is closer to the excitation domain has much more effect than the other one. We also carried out an experiment with the isotropic conductivity tensor of the form $\sigma_{il} = \sigma_{it} = 3{\cdot}10^{-4}$. In this case the excitation wave is not dampened out by the chosen control configuration, due to the domination of the reaction part.

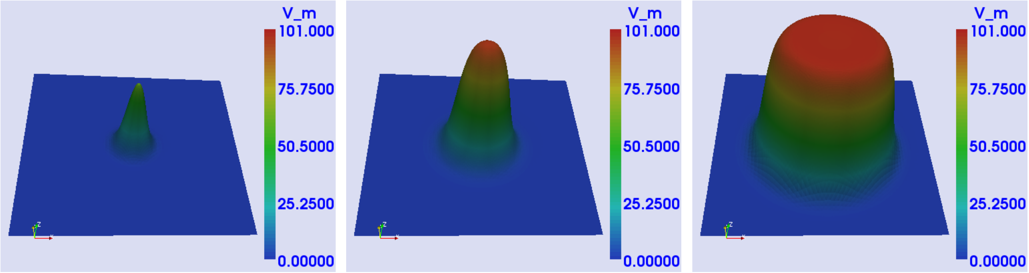

Figure 11: Uncontrolled solution (V_m) at 0.3 msec, 2 msec and 4 msec of simulation time respectively.

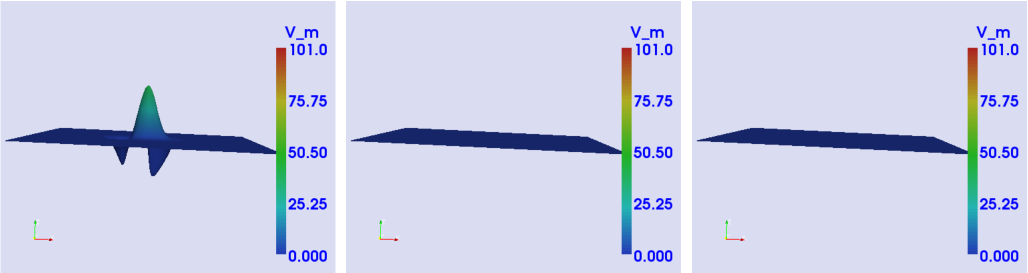

Figure 12: Controlled solution ($(V_m)_{opt}$) at 0.3 msec, 2 msec and 4 msec of simulation time respectively.

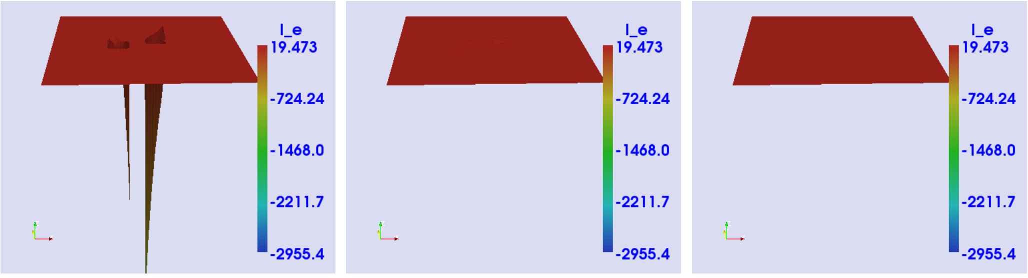

Figure 13: Controlled (I_e) at 0.3 msec, 2 msec and 4 msec of simulation time respectively.

Again we can observe super-linear convergence of the optimization algorithm.

6 Conclusions

We discussed solution strategies based on the Newton method and adaptive grid refinement strategies to solve optimal control of the action potential in cardiac electrophysiology based on the monodomain model. The numerical results show the capability of dampening the excitation wave propagation by properly applying an extra cellular current as a control variable. The presented optimization algorithm exhibits super linear convergence of the inexact Newton method. Moreover the overall performance is more efficient when compared to first order methods, see [13]. The results motivate us to continue our investigations for the bidomain model. Adaptive time- stepping should be considered to further save computational time. We also need to strive for more insight with respect to longer time horizons, with more realistic geometries, and finer meshes.

Acknowledgement

The authors gratefully acknowledge the Austrian Science Foundation (FWF) for financial support under SFB 032, "Mathematical Optimization and Applications in Biomedical Sciences".

References

[1] P. Bastian, M. Blatt, A. Dedner, C. Engwer, R. Klöfkorn, R. Kornhuber, M. Ohlberger, and O. Sander. A generic grid interface for parallel and adaptive scientific computing. Part II: implementation and tests in dune. *Computing*, 82(2):121–138, July 2008.

[2] R. Becker, H. Kapp, and R. Rannacher. Adaptive finite element methods for optimal control of partial differential equations: Basic concept. *SIAM J. Control Optim.*, 39(1):113–132, 2000.

[3] R. Becker and R. Rannacher. An optimal control approach to a posteriori error estimation in finite element methods. *Acta Numerica*, 10(1):1–102, 2001.

[4] Y. Bourgault, Y. Coudiére, and C. Pierre. Existence and uniqueness of the solution for the bidomain model used in cardiac electrophysiology. *Nonlinear Analysis: Real World Applications*, 10(1):458–482, 2009.

[5] E. M. Cherry, H. S. Greenside, and C. S. Henriquez. A space-time adaptive method for simulating complex cardiac dynamics. *Phys. Rev. Lett.*, 84(6):1343–1346, Feb 2000.

[6] L. Dede' and A. Quarteroni. Optimal control and numerical adaptivity for advection-diffusion equations. *M2AN*, 39(5):1019–1040, sep 2005.

[7] P. C. Franzone, P. Deuflhard, B. Erdmann, J. Lang, and L. F. Pavarino. Adaptivity in space and time for reaction-diffusion systems in electrocardiology. *SIAM Journal on Numerical Analysis*, 28(3):942–962, 2006.

[8] C. S. Henriquez. Simulating the electrical behavior of cardiac tissue using the bidomain model. *Crit. Rev. Biomed. Eng.*, 21:1 77, 1993.

[9] M. Hinze and K. Kunisch. Second order methods for optimal control of time-dependent fluid flow. *SIAM J. Control Optim.*, 40(3):925–946, 2001.

[10] M. Hinze, R. Pinnau, M. Ulbrich, and S. Ulbrich. *Optimization with PDE Constraints.* Springer, 2008.

[11] J. Lang. *Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems*, volume 16 of *Lecture Notes in Computational Science and Engineering.* Springer-Verlag, Berlin, 2001.

[12] P. K. Moore. An adaptive finite element method for parabolic differential systems: Some algorithmic considerations in solving in three space dimensions. *SIAM Journal on Scientific Computing*, 21(4):1567–1586, 1999.

[13] C. Nagaiah, K. Kunisch, and G. Plank. Numerical solution for optimal control of mono-domain equations in cardiac electrophysiology. *appear in Computational Optimization and Applications, doi:10.1007/s10589-009-9280-3.*

[14] C. Nagaiah, K. Kunisch, and G. Plank. Numerical solutions for optimal control of monodomain equations in cardiac electrophysiology. In *proceedings of BFG-09 (accepted).*

[15] B. F. Nielsen, T. S. Ruud, G. T. Lines, and A. Tveito. Optimal monodomain approximations of the bidomain equations. *Applied Mathematics and Computation*, 184(2):276–290, 2007.

[16] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer Verlag, New York, second edition edition, 2006.

[17] G. Plank, M. Liebmann, R. W. dos Santos, E. Vigmond, and G. Haase. Algebraic multigrid preconditioner for the cardiac bidomain model. *IEEE Trans Biomed Eng.*, 54(4):585–596, 2007.

[18] R. Plonsey. Bioelectric sources arising in excitable fibers (ALZA lecture). *Ann Biomed Eng*, 16(6):519–46, 1988.

[19] M. Potse, B. Dube, J. Richer, A. Vinet, and R. Gulrajani. A comparison of monodomain and bidomain reaction-diffusion models for action potential propagation in the human heart. *IEEE Transactions on Biomedical Engineering*, 53(12):2425–2435, Dec. 2006.

[20] J. M. Rogers and A. D. McCulloch. A collocation-Galerkin finite element model of cardiac action potential propagation. *IEEE Trans. Biomed. Eng.*, 41:743–757, 1994.

[21] J. A. Trangenstein and C. Kim. Operator splitting and adaptive mesh refinement for the luo-rudy i model. *J. Comput. Phys.*, 196(2):645–679, 2004.

[22] L. Tung. *A bi-domain model for describing ischemic myocardial DC potentials.* PhD thesis, MIT, Cambridge, MA, 1978.

[23] R. Verfürth. A review of a posteriori error estimation and adaptive mesh-refinement techniques. *Wiley & Teubner*, 1996.

[24] E. J. Vigmond, R. Weber dos Santos, A. J. Prassl, M. Deo, and G. Plank. Solvers for the cardiac bidomain equations. *Prog Biophys Mol Biol*, 96(1-3):3–18, 2008.

[25] R. Weber dos Santos, G. Plank, S. Bauer, and E. Vigmond. Parallel multigrid preconditioner for the cardiac bidomain model. *IEEE Trans. Biomed. Eng.*, 51(11):1960–1968, 2004.

[26] O. C. Zienkiewicz and J. Z. Zhu. A simple error estimator and adaptive procedure for practical engineering analysis. *Int. J. Num. Meth. Eng*, 24:337–357, 1987.

A Optimization algorithm

Here is a brief outline of complete a Newton's optimization algorithm, by utilizing the spatial grid adaptivity during the primal and dual solves, which is used to carry out numerical computations.

Algorithm 1 Line search Newton-CG optimization algorithm.

```
primal variables: $v, w$.
dual variables : $p, q$.
repeat
    set $v(0) := v^0, w(0) := w^0$ and solve the primal problem to obtain $v(t), w(t)$ by utilizing the spatial grid adaptation during the intermediate time steps.
    solve the dual problem for $p(t), q(t)$ using terminal conditions $p(T) := 0, q(T) := 0$ by utilizing the spatial grid adaptation during the intermediate time steps.
    update the gradient using the adjoint solution $\hat{J}' = p + \alpha I_{e_k}$.
    solve the system $\hat{J}'' \delta I_e = -\hat{J}'$ by linear conjugate gradient method.
    set step length $\beta_k := 1.0$ and compute optimal $\beta_k$ using backtracking method by checking the strong Wolfe conditions, see [16].
    update $I_{e_{k+1}} := I_{e_k} + \beta_k \delta I_e$ using modified $\beta_k$.
    $k \leftarrow k + 1$.
until $\|\nabla J_k\|_\infty \leq 10^{-3}(1 + |J_k|)$
```