On the performance of anisotropic mesh adaptation for scroll wave turbulence dynamics in reaction-diffusion systems

Youssef Belhamadia, André Fortin, Yves Bourgault

PII: S0377-0427(14)00187-3
DOI: http://dx.doi.org/10.1016/j.cam.2014.04.006
Reference: CAM 9648

To appear in: Journal of Computational and Applied Mathematics

Received date: 28 August 2013
Revised date: 23 January 2014

Please cite this article as: Y. Belhamadia, A. Fortin, Y. Bourgault, On the performance of anisotropic mesh adaptation for scroll wave turbulence dynamics in reaction-diffusion systems, Journal of Computational and Applied Mathematics (2014), http://dx.doi.org/10.1016/j.cam.2014.04.006

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.
On the Performance of Anisotropic Mesh Adaptation for Scroll Wave Turbulence Dynamics In Reaction-Diffusion Systems

Youssef Belhamadia,
Campus Saint-Jean and Mathematical Department,
University of Alberta, Edmonton, Canada.
emails: youssef.belhamadia@ualberta.ca,

André Fortin
Department of Mathematics and Statistics, Laval University, Québec, Canada

Yves Bourgault
Department of Mathematics and Statistics, University of Ottawa, Ottawa, Canada
On the Performance of Anisotropic Mesh Adaptation for Scroll Wave Turbulence Dynamics In Reaction-Diffusion Systems

Youssef Belhamadia, André Fortin, and Yves Bourgault

Abstract

Nonlinear reaction-diffusion systems are widely employed to study the spatio-temporal chaotic behavior that occurs in excitable media such as cardiac tissue where sufficiently strong perturbations can excite nonlinear propagating waves which can form spiral waves in two dimensions or scroll waves in three dimensions. The numerical simulation of these waves calls for grids that are extremely fine over the whole computational domain to accurately predict the trajectories and multiplication of wave fronts and therefore leads to huge computational challenges. Mesh adaptation methods can reduce the number of degrees of freedom required for a given accuracy but they also have a cost and it is not clear if they are competitive with respect to very fine uniform meshes. Previous mesh adaptation techniques applied to spatio-temporal chaotic behavior have been mostly limited to the two-dimensional case. The purpose of this paper is to explore the efficiency of a three-dimensional anisotropic finite element mesh adaptivity for simulating scroll wave turbulence. The computational efficiency of the proposed method is assessed using reference solutions obtained on a uniform refined mesh with more than 44 millions degrees of freedom. The proposed method reduces significantly the number of elements leading to huge saving in memory as well as in computational time. Examples of the dynamics of ventricular fibrillation in cardiac tissue will be presented illustrating the performance of the overall methodology.

1 Introduction

Spiral and scroll wave turbulence dynamics are found in a variety of nonlinear excitable media. The occurrence and multiplication of spiral and scroll waves in such media result in a spatio-temporal chaotic behavior. This behavior has been experimentally observed in different biological, chemical and physical systems including the Belousov-Zhabotinsky reaction (Zhabotinsky [1]), the oxidation of CO on platinum catalysts (Bar et al. [2]), and in electrical membrane excitation in nerve and muscular tissues (Keener [3]). One of the most important example is the propagation of electrical turbulence in the heart which is known as ventricular fibrillation and is the main cause of sudden cardiac death in the world. More details about the nonlinear waves in excitable media can be found in Tyson and Keener [4] and Bernus et al. [5].

From a numerical point of view, most of the spiral and scroll wave turbulence can be reproduced by numerical solution using the monodomain equations, which consists of a nonlinear partial differential equation coupled with an ordinary differential equation. In Barkley et al. [6] and Barkley [7], a modified version of the FitzHugh-Nagumo model (monodomain model) has been proposed to produce spiral waves in excitable media. In Bar and Eiswirth [8], a piecewise linearized FitzHugh-Nagumo model has been employed to produce turbulent wave patterns in reactions similar to the oxidation of CO on platinum surface and numerical simulations have been performed on a mesh with $250 \times 250$ grid points. In Ten Tusscher and Panfilov [9], spiral and scroll wave dynamics in presence of non excitable cells in an excitable media has been studied using the Aliev-Panfilov model. Numerical simulations have been performed using $200 \times 200$ grid points in 2D and $200 \times 200 \times 76$ grid points in 3D. The parameters in the model were adjusted to reproduce key characteristics of cardiac tissue.

Although the turbulence dynamics can be reproduced with a simple two-component reaction-diffusion system, the spatio-temporal chaos leads to a multiplication of wave fronts calling for grids that are uniformly refined over the whole computational domain and therefore requires a huge amount of computational resources. In cardiac tissue, however, the monodomain model assume the same anisotropy ratio for intra and extracellular spaces and therefore the use of a bidomain model, which provides more realistic description of the electrical activity and
represents the transmembrane potential in the intra- and extra-cellular spaces of the cardiac tissue, is crucial. The bidomain model requires an additional elliptic equation with respect to the monodomain equations which makes this model more computationally intensive. The study of scroll wave dynamics with the bidomain model is therefore more complex and faces several computational challenges and this is why previous studies have been mostly limited to the two-dimensional case.

Many methods have been developed to reduce the computational burden associated to the monodomain and bidomain models such as parallel computing techniques, implicit time-stepping discretizations, operator-splitting methods, etc. A more complete review can be found in Sundnes [10] and Linge et al. [11]. More recently, mesh adaptation methods have been introduced for simulating the wave fronts in reaction-diffusion systems. In Bendahmane et al. [12], a finite volume scheme enriched by a fully adaptive multiresolution method is presented for the monodomain and bidomain models. The method was shown to accurately simulate single electrical wave in myocardial tissue. In Cherry et al. [13] and Trangenstein and Kim [14], an adaptive mesh refinement algorithm (AMRA), as developed by Berger and Colella [15], is applied to simulate single propagating pulse in LuoRudy 1 (LR1) and FitzHugh-Nagumo models (monodomain models). ColliFranzone et al. [16] presents an adaptive numerical method in space and time for cardiac reaction-diffusion models. Numerical simulation of the evolution of a regular heartbeat was presented. An enhancement of the adaptive methods by considering anisotropic mesh adaptation techniques, where the mesh elements can be elongated along some appropriate directions, has been proposed for the first time, in the context of electrocardiology, in Belhamadia [17]. This technique consists in locating finer and elongated mesh cells near the front position while a coarser mesh is used away from the front. The method was based on an approximation of a hierarchical error estimator and two-dimensional results were presented. In Belhamadia et al. [18], anisotropic mesh adaptation has been considered in three-dimensional case and the error estimator was based on the definition of edge length using a solution dependent metric. The monodomain model was employed for a single wave to assess the performance of the proposed method. Anisotropic mesh adaptation has been also used in Southern et al. [19] using the bidomain model for simulating a single electrical front in a three dimensional realistic heart geometry.

Most of the previous works on adaptive method are limited to the numerical simulation of a single wave front. To the authors knowledge, very few contributions have been reported in the literature for spiral and scroll wave turbulence. In Cherry et al. [20], the AMRA technique has been applied to two dimensional multiple spiral waves generated by the monodomain model with LR1 while the algorithm performance in the single wave case was presented in Cherry et al. [13]. They report a speedup factor of 20 in the two dimensional case and for a single propagating pulse with an average memory saving factor of 10 when compared to an algorithm using a uniform space-time mesh at the finest resolution. For multiple spiral waves, the speedup factor was around 5 with a comparable reduction in memory. In moving to a domain with more complex dynamics in 2D, then, the speedup was reduced by a factor of 4 and the memory savings by a factor of 2. They concluded that a simulation of more complex dynamics in the three dimensional case with the same spatial and temporal resolution was beyond their group’s computational resources at that time. The adaptive finite volume multiresolution method, as presented in Bendahmane et al. [12] for single cardiac tissue wave, has been applied in Bürger et al. [21] for simulation of the two-dimensional complex dynamic of waves in excitable media and it has been also applied in Bürger and Ruiz-Baier [22] to the simulation of the complex dynamics in electrocardiology by using two-dimensional Aliev-Panfilov and Luo-Rudy II models. Finally, Deufhard et al. [23] show three-dimensional adaptive finite element simulations of ventricular fibrillation dynamics. The monodomain model using Aliev-Panfilov ion kinetics was employed and the numerical simulation was obtained by applying the code KARDOS [24]. Although this paper deals with the three-dimensional cases, KARDOS is limited to unstructured isotropic meshes.

Clearly, the performance of any adaptive method should be assessed with complex wave dynamics. The gain in computational time is not easy to establish, especially in presence of a multiplication of spiral and scroll waves that calls for finer meshes everywhere in the computational domain. To our knowledge, three-dimensional anisotropic adaptive methods for simulating scroll wave turbulence has not been previously reported in the literature and this is the main purpose of this work. An investigation of the performance of anisotropic adaptive finite element method for simulating three-dimensional complex wave dynamics is therefore presented. In addition to the monodomain, the bidomain model will be also employed. Examples of ventricular fibrillation in cardiac tissue and turbulent wave patterns similar to those observed in Belousov-Zhabotinsky reaction will be presented illustrating the performance of the overall methodology.
The remaining part of this paper is organized as follows. The next section is devoted to presenting the bidomain and monodomain models. Section 3 briefly describes the adaptive mesh algorithm, and Section 4 presents three dimensional numerical results showing the accuracy of the proposed method.

2 Mathematical Model

The bidomain model consists of a system of two nonlinear reaction-diffusion equations for the intracellular and extracellular potentials, \( \phi_i \) and \( \phi_e \). This model is commonly formulated in terms of the transmembrane potential, \( V_m = \phi_i - \phi_e \) and the extracellular potentials \( \phi_e \). The non-linearity arises from the ionic model which is, in this paper, described by a nonlinear ordinary differential equation. The bidomain model can be stated as follows:

\[
\begin{aligned}
\frac{\partial V_m}{\partial t} - \nabla \cdot (G_i \nabla V_m) &= \nabla \cdot (G_i \nabla \phi_e) + I_{ion}(V_m, W) + I_s, \\
\nabla \cdot ((G_i + G_e) \nabla \phi_e) &= -\nabla \cdot (G_i \nabla V_m), \\
\frac{\partial W}{\partial t} &= g(V_m, W),
\end{aligned}
\]  

(1)

where \( W \) is the recovery variable, \( G_i \) and \( G_e \) are the symmetric intra- and extra-cellular conductivity tensors, respectively. \( I_s \) is the current due to an external stimulus and the functions \( I_{ion}(V_m, W) \) and \( g(V_m, W) \) represent the current due to the ionic exchange and their definition depends on the ionic model chosen. In the literature, many ionic models are employed, in particular two-variable models such as those proposed by Barkley [7], and Panfilov [25, 26], etc. In this paper, Aliev-Panfilov model (see Aliev and Panfilov [27]) is considered and it consists in the following equations:

\[
\begin{aligned}
I_{ion}(V_m, W) &= kV_m(V_m - a)(1 - V_m) - V_mW, \\
g(V_m, W) &= \left(\epsilon + \frac{aW}{g_e + V_m}\right)(-W - kV_m(V_m - b - 1)).
\end{aligned}
\]  

(2)

One popular way of reducing the bidomain computational time is to reduce the two by two set of partial differential equations to a scalar partial differential equation. This leads to the monodomain model that is obtained under the assumption of equal anisotropy ratio, \( G_i = \lambda G_e \), which is generally unrealistic. The final monodomain model can be stated as follows:

\[
\begin{aligned}
\frac{\partial V_m}{\partial t} - \nabla \cdot (G \nabla V_m) &= I_{ion}(V_m, W), \\
\frac{\partial W}{\partial t} &= g(V_m, W),
\end{aligned}
\]  

(3)

The conductivity tensor \( G \) can be considered as \( \frac{G_i}{1 + \lambda} \). Examples of the monodomain model (3) representing the dynamic of scroll wave will be also considered to illustrate the mesh adaptivity. In Bürger et al. [21], mesh adaptivity was employed for the monodomain model with ionic models leading to spatio-temporal chaos. Similar ionic models will be employed in this paper. In addition to the classical Aliev-Panfilov model (2), a modified version of a piecewise linear FitzHugh-Nagumo model will be used as well (see Bar and Eiswirth [8], and Pandit et al. [28])

\[
\begin{aligned}
I_{ion}(V_m, W) &= kV_m(1 - V_m)(V_m - \frac{W + k}{a}), \\
g(V_m, W) &= f(V_m) - W,
\end{aligned}
\]  

(4)

where

\[
f(V_m) = \begin{cases} 
0, & \text{if } V_m < \frac{1}{3}, \\
1 - cV_m(V_m - 1)^2, & \text{if } \frac{1}{3} \leq V_m \leq 1, \\
1, & \text{if } V_m > 1.
\end{cases}
\]
The discretization of the different models is obviously crucial and accuracy is of extreme importance in such calculations. In all numerical simulations, a quadratic finite element \( P_2 \) for spatial discretization and a fully implicit backward second order scheme (Gear) for the time derivative discretization are employed. The reader is referred to Belhamadia [17] for a detailed comparison between the different spatial discretizations and time-stepping schemes. The reader is referred to Chamberland et al. [29] as well, where the performance of various finite element discretizations is analysed in terms of computational cost and memory requirements. It is shown that for a given level of accuracy, quadratic elements are much more efficient than linear ones. This is why we favor quadratic tetrahedral elements though our mesh adaptation method works equally well for both linear and quadratic elements.

Starting from \( V_m^{n-1} \) and \( W^{n-1} \) at time \( t^{n-1} \) and from \( V_m^n \) and \( W^n \) at time \( t^n \), Gear scheme gives:

\[
\frac{\partial V_m}{\partial t}(t^{n+1}) \approx \frac{3V_m^{(n+1)} - 4V_m^{(n)} + V_m^{(n-1)}}{2\Delta t}
\]

and similarly for \( \frac{\partial W}{\partial t}(t^{n+1}) \). For more details about a comparison between different time-stepping schemes and spatial discretizations, the reader is referred to Belhamadia [17]. The variational formulation of the system of nonlinear equation (1) is straightforward and obtained by multiplying this system by test functions \((\psi_v, \psi_\phi, \psi_w)\) such that:

\[
\begin{align*}
\int_{\Omega} 3V_m^{(n+1)} - 4V_m^{(n)} + V_m^{(n-1)} \, \psi_v \, d\Omega + \int_{\Omega} G_i \nabla V_m^{(n+1)} \cdot \nabla \psi_v \, d\Omega \\
\quad + \int_{\Omega} G_i \nabla \phi_e^{(n+1)} \cdot \nabla \psi_v \, d\Omega = \int_{\Omega} I_{con}(V_m^{(n+1)}, W^{(n+1)}) \psi_v \, d\Omega,
\end{align*}
\]

\[
\begin{align*}
- \int_{\Omega} (G_i + G_e) \nabla \phi_e^{(n+1)} \cdot \nabla \psi_\phi \, d\Omega = \int_{\Omega} G_e \nabla V_m^{(n+1)} \cdot \nabla \psi_\phi \, d\Omega,
\end{align*}
\]

\[
\int_{\Omega} 3W^{(n+1)} - 4W^{(n)} + W^{(n-1)} \, \psi_w \, d\Omega = \int_{\Omega} g(V_m^{(n+1)}, W^{(n+1)}) \psi_w \, d\Omega.
\]

The variational formulation for the monodomain model (3) is obtained similarly as (5) and is not presented in this paper to avoid repetition. Newton’s method is used to solve the non linear system above at each time step. Linear system resulting from Newton’s method are solved by iterative methods, GMRES solver (Saad [30]) from the PETSc library (Balay et al. [31]).

\section{Adaptive Method}

The main goal of the adaptive method is to reduce the spatial mesh size as well as the computational time while maintaining and even improving the accuracy of the solution. Many strategies have been introduced in the context of simulating the cardiac electrical activity. In Deufhard et al. [23] and Colli Franzone et al. [16], an adaptive Rothe method is used. These authors first discretize the system of PDEs in time using a linearly implicit Rosenbrock discretization with stepsize control, then in space with an adaptive multilevel finite element method controlled by a posteriori local error estimators. In Belhamadia [17], an adaptive time dependent algorithm was developed for the two-dimensional simulation of electrical waves in the heart. The method is based on a hierarchical error estimator. A three dimensional adaptive method based on a definition of edge lengths using a solution dependent metric is employed in Belhamadia et al. [18] for simulating regular wave front in electrocardiology.

In this work, the adaptive technique is based on a definition of edge lengths using a solution dependent metric for both two- and three-dimensional cases. This technique is similar to the method described in Belhamadia et al. [32]. This method has been abundantly described in the literature and the reader is also referred to Habashi et al. [33], Ait Ali Yahia et al. [34], Dompièrre et al. [35] and Hecht and Mohammadi [36] for a complete presentation.
3.1 Anisotropic mesh adaptation

It is known, see [32], that an optimized mesh must satisfy:

\[(\mathbf{x}_j - \mathbf{x}_i)^T A(\mathbf{x}_i) (\mathbf{x}_j - \mathbf{x}_i) = C\]  

(6)

for every edge of the mesh, where \( \mathbf{x}_i \) and \( \mathbf{x}_j \) are two vertices and \( A \) is the Hessian matrix that can be reconstructed from the linear approximation solution \( u_h \) in a least square sense using the method described in [37]. Supposing now that \( A \) is positive definite, Eq. (6) defines a new (Riemannian) norm:

\[||v||_A = (v^T A(P)v)^{1/2} = (A(P)v \cdot v)^{1/2},\]  

(7)

and thus the condition for an optimized mesh becomes:

\[||\mathbf{x}_i - \mathbf{x}_j||_A = C\]  

(8)

for every edge of the mesh and for some given target edge length \( C \). The main objective of mesh adaptation is therefore to provide a new mesh with edge length \( C \) satisfying the equation (8). The anisotropic mesh adaptation procedure is based on edge refinement, edge swapping, vertex suppression and vertex displacement. In general, edge refinement and vertex suppression are used in order to control edge lengths while vertex displacement and edge swapping are used to improve the quality of the elements i.e. to avoid degenerate tetrahedra. As will be seen, this procedure is enough to provide strongly anisotropic meshes and determine if a given local operation is needed based on the following:

- **Edge refinement**
  
  To decide if an edge between node \( \mathbf{x}_i \) and \( \mathbf{x}_j \) will be halved, the computation is performed at both extremities and averaged to give the value on the edge:

\[
\frac{(A(\mathbf{x}_i)(\mathbf{x}_j - \mathbf{x}_i))^{1/2} + (A(\mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j))^{1/2}}{2}.
\]  

(9)

If this value is larger than the prescribed value \( C \), the edge is halved by creating a new vertex at mid-edge. If the given edge is on the boundary, the new vertex must be projected on the geometry of the domain.

- **Vertex suppression**
  
  The edges of the mesh are swept and their length computed according to Eq. (8). If the length of an edge is too small (i.e. the error is small) then one of its vertex is removed creating a “hole” in the mesh. This “hole” is then remeshed by adding edges between some of the remaining nodes. This can be done in different ways and the precise choice of the nodes to be joined is not important at this stage. The process of edge swapping (see next) will correct this choice if needed.

- **Edge swapping**
  
  To perform this local operation the quality of the elements is introduced by the following criteria

\[
\frac{(\det A)^{1/2} V(T)}{\frac{1}{2} ||\mathbf{a}_1||_A ||\mathbf{a}_2||_A ||\mathbf{a}_3||_A}.
\]  

(10)

where \( \mathbf{a}_1 \), \( \mathbf{a}_2 \) and \( \mathbf{a}_3 \) are three vectors issued from a vertex \( P \) and \( T \) is the tetrahedron generated by these vectors with volume \( V(T) \). The ratio (10) measures the discrepancy between a given tetrahedron and a rectangular tetrahedron, but using the new metric. The maximum value of this ratio is 1 and it is a good indication of the regularity of the tetrahedron.

The quality of a tetrahedron is determined by the ratio (10) where it is computed (and denoted \( Q_i \)) at each vertex \( P_i \), and the geometric mean is taken:

\[Q(T) = \sqrt[4]{Q_1 Q_2 Q_3 Q_4}.
\]  

(11)

The quality of the elements must be as far as possible from 0 to avoid degenerate tetrahedra.
The edge swapping operation is mainly used to improve the quality of the elements defined by Eq. (11). The elements are first swept and their quality $Q(T)$ is computed. For a given tetrahedron, if $Q(T) < Q_d$, where $Q_d$ is a prescribed minimum value, then the edges of that element are tagged. Then all tagged edges are swept, the shell around each edge is built and the average quality of the elements in that shell is computed. Then, the shell obtained by swapping the edge is built and the average quality of the resulting new elements is computed. If the new average is larger than the initial one, the edge is swapped. Otherwise, the edge is put back in its initial position.

- **Vertex displacement**

A vertex can be moved inside its shell by considering that the edges form a network of springs whose stiffness is proportional to the edge error. The vertex is then moved by trying to minimize the total energy of the system. A complete description of the method and minimization technique is given in Habashi et al. [33].

The sequence of edge refinement, edge swapping, vertex suppression and vertex displacement is repeated 3 to 5 times. Experience shows that after this number of iterations, the meshes do not evolve significantly and the edges have more or less the prescribed length $C$. Minimum and maximum lengths must also be added to avoid regions which are too coarse or too refined (in the vicinity of a singularity for instance).

### 3.2 Adaptive algorithm for time dependent problems

The overall adaptive algorithm for time dependent problems is the following.

1. Start from the solutions $V_m^{(n-1)}$, $V_m^{(n)}$, $W^{(n-1)}$, $W^{(n)}$, $\phi_\Sigma^{(n-1)}$ and $\phi_\Sigma^{(n)}$ and a mesh $\mathcal{M}^{(n)}$ at time $t^{(n)}$;

2. Solve the system (3) on mesh $\mathcal{M}^{(n)}$ to obtain a first approximation of the solutions (denoted $V_m^{(n+1)}$, $W^{(n+1)}$ and $\phi_\Sigma^{(n+1)}$) at time $t^{(n+1)}$;

3. Adapt the mesh starting from mesh $\mathcal{M}^{(n)}$ and the solution dependent metric calculated from the solutions $V_m^{(n-1)}$, $V_m^{(n)}$, $W^{(n-1)}$, $W^{(n)}$, $\phi_\Sigma^{(n-1)}$, $\phi_\Sigma^{(n)}$ and $\phi_e^{(n+1)}$ to obtain a new mesh $\mathcal{M}^{(n+1)}$;

4. Reinterpolate $V_m^{(n-1)}$, $V_m^{(n)}$, $W^{(n-1)}$, $W^{(n)}$, $\phi_\Sigma^{(n-1)}$ and $\phi_\Sigma^{(n)}$ on the mesh $\mathcal{M}^{(n+1)}$;

5. Solve the system (3) on the mesh $\mathcal{M}^{(n+1)}$ for $V_m^{n+1}$, $W^{n+1}$ and $\phi_e^{n+1}$.


Some modifications have been introduced in the way to adapt the mesh on the different variables with respect to what was described in Belhamadia et al. [18]. Our time-discretization scheme requires an accurate representation of the different solutions at three different time steps $t^{(n-1)}$, $t^{(n)}$, and $t^{(n+1)}$. Adapting the mesh using all the different numerical solutions at each time step, as presented in step 3, is therefore necessary. Another difficulty comes from the quadratic discretization ($P_2$) of each of these variables. Mesh adaptation based on the notion of metric is a priori valid only for linear discretizations ($P_1$). This difficulty can be overcome by adapting on the gradient of the different variables (see Chamberland and Fortin [38]). But this multiply by 3 the number of metrics that have to be constructed. All these metrics should then be intersected as proposed in Alauzet and Frey [39] in order to provide a common metric on which the mesh is adapted taking into account the errors on all variables. Step 3 indicates that 9 variables should be taken into account and therefore 27 metrics would need to be constructed. This would be extremely costly. To reduce the computational cost associated to adaptation, one can adapt on a linear combination of the different solutions. In our experience, the three variables:

$$
\frac{V_m^{(n-1)} + V_m^{(n)} + V_m^{(n+1)}}{3}, \quad \frac{W^{(n-1)} + W^{(n)} + W^{(n+1)}}{3}, \quad \frac{\phi_\Sigma^{(n-1)} + \phi_\Sigma^{(n)} + \phi_e^{(n+1)}}{3}
$$

provide satisfactory results but still requires the construction of 9 different metrics (instead of 27).
4 Numerical Results

Three cases are presented to illustrate the performance and the efficiency of the proposed method in capturing complex scroll wave dynamics. The first case uses a modified version of the FitzHugh-Nagumo model to simulate spatio-temporal chaotic systems similar to what is observed in the Belousov-Zhabotinsky reaction. Quantitative results are presented by comparing the numerical solutions obtained with various structured and adapted meshes to a reference solution obtained on an extremely fine mesh. The second case uses the Aliev-Panfilov ionic model with the monodomain equations. We will show that mesh adaptation allows the preservation of scroll wave filaments. The last case is devoted to the bidomain equations coupled with the Aliev-Panfilov model and quantitative results are presented for the simulation of spiral and scroll wave dynamics in electrocardiology.

4.1 Monodomain with FitzHugh-Nagumo Model

It is well known that spatio-temporal chaotic systems require very fine meshes. In Bar and Eiswirth [8], a grid with $256 \times 256$ points is used to simulate the turbulence due to spiral breakup in a continuous excitable media using a modified version of the FitzHugh-Nagumo model. In Morgan et al. [40], a control study for scroll wave turbulence is presented using a structured mesh with $90 \times 90 \times 90$ elements. The only adaptive simulations of wave turbulence dynamics (to the authors knowledge) can be found in Bürger et al. [21] using a finite volume technique and only in the two-dimensional case. The three dimensional case is still a challenging task and our goal is to show the advantages of the anisotropic adaptive technique presented in the previous section for scroll wave turbulence dynamics.

The three-dimensional scroll wave is obtained with a technique similar to the one described in Ezscroll software by Barkley [7], Dowle et al. [41]. More specifically, we set the following initial conditions

\[
V_m(x, 0) = \begin{cases}
1, & \text{if } (x < 30) \text{ and } (y > 35), \\
1, & \text{if } (x \geq 30) \text{ and } (y < 35), \\
0, & \text{otherwise}.
\end{cases}
\]

\[
W(x, 0) = \begin{cases}
0.375, & \text{if } (x < 30) \text{ and } (z > 30), \\
0.375, & \text{if } (x \geq 30) \text{ and } (z < 30), \\
0, & \text{otherwise}.
\end{cases}
\]

The computational domain is $[0, 60] \times [0, 60] \times [0, 60]$. Homogeneous Neumann boundary conditions are imposed on all sides and the following parameters have been used (see Eq. (2)):

\[
\begin{array}{cc}
a = 0.84 & b = 0.08 \\
k = 14 & c = 6.75 \\
G = 1 & \Delta t = 0.2
\end{array}
\]

As discussed in Bar and Eiswirth [8], the parameter $k$ is decisive in the behaviour of the solution. For a broad range of values of $b$ ($b \leq 0.18, a = 0.84$) and $k = 11.5$ the modified FitzHugh-Nagumo model produces scroll waves. The initial scroll wave for this test will therefore be obtained with $k = 11.5$ using the above initial conditions on an extremely fine mesh ($16,464,000$ tetrahedral elements leading to $44,376,082$ dof) which is reinterpolated on all other meshes to ensure the same starting wave.

As also discussed in Bar and Eiswirth [8], for $k \geq 14$, the scroll wave begins to break up after some transient rotations and this is illustrated in Figure 1 for $k = 14$. This figure illustrates the variable $V_m$ and its corresponding adapted meshes at different times. It is clear that elongated elements are obtained at the appropriate position and the multiplication of the wave fronts has been adequately predicted.

As can also be seen in Figure 1, the wave travels across the whole computational domain and it will be interesting to compare the solutions on adapted meshes with those obtained on uniformly refined grids. In absence of an exact (analytical) solution, the numerical solutions obtained on various structured and adapted
meshes will be compared to a reference solution, \( V_{m}^{ref} \), obtained on an extremely fine mesh of 16 464 000 tetrahedral elements which lead to more than 44 million degrees of freedom since we use quadratic discretization for both variables \( V_m \) and \( W \). The following error is used to compare numerical solutions denoted \( V_{m}^{h} \), with the reference solution, \( V_{m}^{ref} \), at time \( t_n \):

\[
E_{2} = \frac{\|V_{m}^{h} - V_{m}^{ref}\|_2}{\|V_{m}^{ref}\|_2}
\]

where

\[
\|V_{m}^{h} - V_{m}^{ref}\|_2 = \left( \frac{1}{d} \sum_{i=1}^{m} |V_{m,i}^{h} - V_{m,i}^{ref}|^2 \right)^{1/2}
\]

d is the dimension of the space, and \( V_{m,i}^{h} \) is the solution \( V_{m}^{h} \) reinterpolated on mesh that leads to \( m = 44 376 082 \) dof.

Table 1 illustrates the error \( E_{2} \) at three different times \( t = 8, t = 16 \) and \( t = 24 \), respectively. These times corresponds to the solutions illustrated in Figure 1. The errors are obtained using three uniform meshes with 162 000, 750 000, and 2 058 000 tetrahedral elements, which lead, respectively, to 453 962, 2 060 602 and 5 606 442 degrees of freedom since we use quadratic discretization for both variables \( V_m \) and \( W \), and are compared with the errors obtained with the adapted meshes using different values of edge length \( C \) described in Section 3.1. The number of tetrahedra, number of dof, and the CPU time indicated in the table for the adapted meshes are averages over all the time steps. The averaged total CPU time per time step is also indicated corresponding to steps 1 to 6 in the algorithm presented in section 3.2. We have also included the CPU time for mesh adaptation only (steps 3 and 4 in the algorithm) to indicate its relative weight. All the numerical simulations have been obtained on a single processor computer except the reference solution which has been computed on a 256 processor parallel computer. As can be seen, at time \( t = 8 \) the scroll wave is on its first rotation and as presented in Table 1, the uniform mesh with 5 606 442 dof performs relatively well in terms of the error \( E_{2} \). However, as the scroll wave begins to break up (see Table 1 at \( t = 16 \) and \( t = 24 \)), the adapted meshes perform much better than the uniform meshes in terms of the both the error \( E_{2} \) and the computational time. It is also important to mention that solving the monodomain model with 5 606 442 dof requires a huge amount of memory and that the adaptive technique is much more efficient on that respect.

<table>
<thead>
<tr>
<th>Structured Meshes</th>
<th>Nb. of dof</th>
<th>Nb. of tetrahedra</th>
<th>( E_{2}(t=8) )</th>
<th>( E_{2}(t=16) )</th>
<th>( E_{2}(t=24) )</th>
<th>Total CPU time per step</th>
<th>CPU time per step for adaptation only</th>
</tr>
</thead>
<tbody>
<tr>
<td>453 962</td>
<td>162 000</td>
<td>0.12</td>
<td>0.53</td>
<td>0.58</td>
<td>1.76min</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>2 060 602</td>
<td>750 000</td>
<td>0.02</td>
<td>0.24</td>
<td>0.33</td>
<td>10.14min</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>5 606 442</td>
<td>2 058 000</td>
<td>0.007</td>
<td>0.12</td>
<td>0.19</td>
<td>24min</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Adapated Meshes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>396 000</td>
<td>142 000</td>
<td>0.014</td>
<td>0.10</td>
<td>0.13</td>
<td>14min</td>
<td>6.52min</td>
<td></td>
</tr>
<tr>
<td>535 000</td>
<td>192 000</td>
<td>0.009</td>
<td>0.07</td>
<td>0.09</td>
<td>18.45min</td>
<td>8.11min</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Comparison of the error \( E_{2} \), number of dof, and number of tetrahedra obtained with the adapted and uniform meshes. Results obtained with the monodomain equations coupled with FitzHugh-Nagumo model at time \( t = 8, t = 16 \) and \( t = 24 \)

### 4.2 Monodomain with Aliev-Panfilov Model

The Aliev-Panfilov model was introduced in 1996 to reproduce more realistic shapes of the cardiac action potential and to reproduce the APD restitution characteristic observed in the experiments (see Aliev and Panfilov [27]). This model presents stiffer depolarization and repolarization fronts compared to FitzHugh-Nagumo model and therefore the numerical results are more dependent on the mesh size especially for three dimensional scroll waves. An adaptive method for the monodomain equations with the Aliev-Panfilov model has been employed in Bürger et al. [21] and in Bürger and Ruiz-Baier [22] for the simulation of the two-dimensional complex dynamic waves
in excitable media and in electrocardiology, respectively. The three dimensional case was considered in Deufhard et al. [23] using adapted but isotropic meshes. The goal of this section is to present a three dimensional anisotropic adaptive method in similar conditions.

The computational domain is \([0, 200] \times [0, 200] \times [0, 200]\) and homogeneous Neumann boundary conditions are imposed on all sides. The initial scroll wave is obtained with similar initial conditions presented in the previous section and then it is broken with the following external stimulus \(I_s\):

\[
I_s = \begin{cases} 
2, & \text{if } \sqrt{(x - 85)^2 + (y - 110)^2} \leq 2, \quad (z \geq 100) \text{ and } 0 \leq t \leq 2, \\
2, & \text{if } \sqrt{(z - 110)^2 + (y - 105)^2} \leq 2, \quad (x \leq 100) \text{ and } 0 \leq t \leq 2, \\
0, & \text{otherwise.}
\end{cases}
\]

The following parameters have been used (see Eq. (2)):

\[
\begin{array}{c|c|c|c}
\text{Parameter} & a & b & G \\
\hline
\alpha & 0.1 & 0.1 & 1 \\
\epsilon & 0.002 & 0.3 & 0.3 \\
\mu_1 & 0.135 & 0.3 & 0.3 \\
\mu_2 & 0.3 & 0.3 & 0.3 \\
\end{array}
\]

Figure 2 shows the time evolution of the solution \(V_m\) and the corresponding adapted meshes. In spite of the breakup of the wave, the number of elements remains small with an average of 500,000 dof. The meshes are refined only in the vicinity of the front position while keeping sufficient resolution in other regions.

To show the accuracy of the solutions on adapted meshes compared to these on uniform ones, the adapted solution with an average of 500,000 dof is compared to the solution \(V_m\) obtained with a uniform mesh using 5,606,442 dof. These two solutions have been obtained with similar computational times on a single processor computer. The cross section of the solutions at \(Z = 190\) and at three different times \((t = 24, t = 54, t = 66)\) are presented in Figure 3. The first column contains the solutions obtained on the coarse uniform mesh, the second the solutions with the adapted meshes, while the last column is for the reference solution obtained with an extremely fine uniform mesh with 44,376,082 dof. In addition to the gains in terms of memory by using the adaptive mesh, figure 3 clearly shows the capacity of the adaptive method to preserve the wave front position. We have investigated multiple cross sections of the computational domain and the conclusions remain the same, which shows that the adaptive mesh preserve the scroll wave filament. This result clearly shows that 5,606,442 dof are still not sufficient to obtain an acceptable wave in the presence of scroll wave turbulence which is not the case for a single regular wave.

Finally, Figure 4 shows a comparison of the isovalue of \(V_m = 0.5\) on the cross section \(Z = 190\) obtained with the adapted mesh (red curves) and uniform mesh (green curves) compared to the same isolines of the reference solution (black curves). Once again, the front position is well captured and the solution with the adapted meshes is much more accurate than the one obtained with the uniform mesh obtained with similar computational time.

### 4.3 Bidomain Equations with Aliev-Panfilov Model

It is generally accepted that the bidomain equations provides a more realistic description of the electrical activity as it provides the electrical potential in the intra- and extra-cellular spaces. However, this model is computationally challenging. Three-dimensional quantitative results for the bidomain model is not yet feasible as it requires massive computational resources. For this reason, the quantitative results presented in this section are obtained only in the two-dimensional case.

The computational domain is \([0, 200] \times [0, 200]\) and homogeneous Neumann boundary conditions are imposed on all sides. The initial condition is a spiral wave obtained on an extremely fine mesh and is broken with the external stimulus \(I_s\):

\[
I_s = \begin{cases} 
1, & \text{if } \sqrt{(x - 100)^2 + (y - 100)^2} \leq 2 \text{ and } 0 \leq t \leq 2, \\
0, & \text{otherwise.}
\end{cases}
\]

The intra- and extra-cellular conductivity tensors are:

\[
G_i = \begin{pmatrix} 3 & 0 \\ 0 & 0.32 \end{pmatrix} \quad \text{and} \quad G_e = \begin{pmatrix} 2 & 0 \\ 0 & 1.24 \end{pmatrix}
\]
The following parameters values have been used (see Eq. (2)):

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a)</td>
<td>0.15</td>
</tr>
<tr>
<td>(b)</td>
<td>0.15</td>
</tr>
<tr>
<td>(\epsilon)</td>
<td>0.001</td>
</tr>
<tr>
<td>(k)</td>
<td>8</td>
</tr>
<tr>
<td>(\mu_1)</td>
<td>0.13</td>
</tr>
<tr>
<td>(\mu_2)</td>
<td>0.3</td>
</tr>
<tr>
<td>(\Delta t)</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Table 2 illustrates the relative error \(E_2\) at two different times \((t = 60\) and \(t = 100\)). The errors are obtained using three uniform meshes with 39 200, 80 000, and 156 800 triangles, which lead, respectively, to 236 883, 482 203 and 944 163 degrees of freedom since we use quadratic discretization for all the variables \(V_m\), \(\phi_e\) and \(W\), and are compared with the error on the adapted meshes using an average of 173 661 dof. The reference solution was again obtained on a very fine mesh using \(m = 7 689 603\) dof \((1 280 000\) triangles). As can be seen, the adapted meshes perform much better than the uniform ones both in terms of the error and the computational time for a given accuracy. The gain in memory is also very significant as the adapted meshes require on average only 173 661 dof \((14 561\) triangles). The time evolution of the solution \(V_m\) and the corresponding adapted mesh are presented in Figure 5. The mesh is clearly refined in the vicinity of the front position leading to accurate numerical solutions.

<table>
<thead>
<tr>
<th>Meshes</th>
<th>Nb. of dof</th>
<th>Nb. of triangles</th>
<th>(E_2(t = 60))</th>
<th>(E_2(t = 100))</th>
<th>Total CPU time per step</th>
<th>CPU time per step for adaptation only</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform</td>
<td>236 883</td>
<td>39 200</td>
<td>0.434</td>
<td>0.644</td>
<td>1.65 min</td>
<td>-</td>
</tr>
<tr>
<td>Uniform</td>
<td>482 203</td>
<td>80 000</td>
<td>0.199</td>
<td>0.339</td>
<td>4.61 min</td>
<td>-</td>
</tr>
<tr>
<td>Uniform</td>
<td>944 163</td>
<td>156 800</td>
<td>0.072</td>
<td>0.112</td>
<td>9.47 min</td>
<td>-</td>
</tr>
<tr>
<td>Adapted</td>
<td>173 661</td>
<td>14 561</td>
<td>0.057</td>
<td>0.085</td>
<td>5.28 min</td>
<td>1.45 min</td>
</tr>
</tbody>
</table>

Table 2: Comparison of the error \(E_2\), number of dof, and number of triangles obtained with the adapted and uniform meshes. Results for multiple waves at time \(t = 60\) and \(t = 100\) obtained with the bidomain equations coupled with Aliev-Panfilov model.

As mentioned before, many adaptive methods have been developed in the context of electrophysiology with a single propagating wave. To complete our analysis, we now present a comparison between single and multiple propagating waves. Similar results will now be presented in the case of a single wave. This wave is obtained by using the following initial condition:

\[
V_m(x, t) = \begin{cases} 
1 & \text{if } \sqrt{(x - 100)^2 + (y - 100)^2} < 10 \\
0 & \text{if } \sqrt{(x - 100)^2 + (y - 100)^2} \geq 10,
\end{cases} \quad W(x, t) = 0, \quad \phi_e(x, t) = 0.
\]

The time evolution of the solution \(V_m\) and the corresponding adapted meshes are presented in Figure 6. Table 3 presents a comparison between adaptive and uniform meshes. As can be seen, adapted anisotropic meshes in the case of single wave perform with very high speed and accuracy compared to uniform meshes. It is also important to mention that uniform meshes perform better for single wave propagation. For instance, at time \(t = 100\) the error \(E_2\) decreases from 0.112 for multiples waves, (see table 2), to 0.032 for single wave (see table 3), with similar computational times for the solutions obtained with 944 163 dof. The adapted meshes perform way faster, at time \(t = 100\) for instance, the error \(E_2\) decreases from 0.085 for multiple waves (see table 3) to 0.018 for single wave (see table 3) and the computational time decreases from 5.28 min per time step for multiple waves to 0.90 min per time step in case of single wave propagation.

This comparison between single and multiple propagating waves confirms that the gain in computational time and the number of mesh elements is not always obvious for solving complex waves dynamic and the performance of any adaptive method should be assessed with spiral and scroll wave turbulence dynamics which was presented in this paper.

5 Conclusions

The performance of an anisotropic time-dependent adaptive method for scroll wave turbulence in reaction-diffusion systems was analysed. The monodomain and bidomain models have been investigated using two different...
### Table 3: Comparison of the error $E_2$, number of dof, and number of triangles obtained with the adapted and uniform meshes. Results for single wave at time $t = 100$ obtained with the bidomain equations coupled with Aliev-Panfilov model

<table>
<thead>
<tr>
<th>meshes</th>
<th>Nb of dof</th>
<th>Nb of triangles</th>
<th>$E_2(t = 100)$</th>
<th>Total CPU time per step</th>
<th>CPU time per step for adaptation only</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform</td>
<td>236 883</td>
<td>39 200</td>
<td>0.244</td>
<td>1.29min</td>
<td>-</td>
</tr>
<tr>
<td>Uniform</td>
<td>482 203</td>
<td>80 000</td>
<td>0.102</td>
<td>3.95min</td>
<td>-</td>
</tr>
<tr>
<td>Uniform</td>
<td>944 163</td>
<td>156 800</td>
<td>0.032</td>
<td>8.71min</td>
<td>-</td>
</tr>
<tr>
<td>Adapted</td>
<td>43 767</td>
<td>3 675</td>
<td>0.018</td>
<td>0.90min</td>
<td>0.28min</td>
</tr>
</tbody>
</table>

ionic models. Despite the fact that the scroll wave turbulence creates multiple wave fronts, the anisotropic mesh adaptive method allows a great reduction of the number of elements and of the required computational resources. Quantitative results were also presented to show the performance of the adaptive meshes in comparison with uniform ones. Although this paper shows a considerable gain in computational resources, some electrocardiology applications may require a parallelization of the adaptive mesh algorithms. The parallelization of our adaptive methods is almost completed and the preliminary results are extremely promising (see Tye Gingras [42]). This will however be the object of a future work. The interested reader is also referred to Southern et al. [43] for a similar discussion.

### Acknowledgements

The authors wish to acknowledge the financial support of National Science and Engineering Research Council (NSERC) and Heart and Stroke Foundation of Canada (HSFC).

Computations were performed on the supercomputer (Colosse) at University Laval with the help of Calcul Québec and Compute Canada. The operation of Colosse is partially funded by the Canadian Foundation for Innovation (CFI), the National Science and Engineering Research Council (NSERC), NanoQuébec, and the Fonds Québécois de Recherche sur la Nature et les Technologies (FQRNT).

### References


Figure 1: Transition from rotating spiral wave to turbulent wave patterns for the monodomain equations with FitzHugh-Nagumo model
Figure 2: Time evolution of $V_m$: monodomain equations with Aliev-Panfilov model
Figure 3: Cross section of the solutions at $Z = 190$: uniform mesh with 5 606 442 dofs (left column), adapted meshes with $\sim 500 000$ dofs (middle) and reference solution with 44 376 082 dofs (right column)
Figure 4: Comparison of isovalue of $V_m = 0.5$ on the cross section $Z = 190$: adapted mesh (red), uniform mesh (green) and reference solution (black)
Figure 5: Time evolution of $V_m$: two-dimensional bidomain equations with Aliev-Panfilov model
Figure 6: Time evolution of $V_m$: two-dimensional bidomain equations with Aliev-Panfilov model for single wave