TECHNISCHE HOCHSCHULE NÜRNBERG
GEORG SIMON OHM

Georg-Simon-Ohm-University of Applied Sciences Nuremberg

Report of the application project at the Faculty of AMP

Simulation of a medical therapy method with finite elements

Martin Michel

Keßlerplatz 12

DE-90489 Nuremberg

Advisor: Prof. Dr. rer. nat. Tim Kröger
Advisor: Prof. Dr. rer. nat. habil. Jörg Steinbach
Advisor: Prof. Dr. rer. nat. Thomas Lauterbach

Nuremberg, 01. January 1900

Contents

List of Figures

List of Tables

Listings

1. Introduction

Lets talk about:
- Medical Treatment of Tumor
- Radio frequency ablation
- Why RFA Simulation is important
- Motivation
- This project in General

2. Computer-aided simulation of radio frequency ablation with finite elements

2.1. About Errors in simulations and numerical approaches

- see TUM dissertation
- There are different sources for errors following the simulation from the line from the real problem down to the discrete solution
- Idealization error: discrepancy between reality and the idealized reality and the idealized constitutive laws and boundary conditions -> Systems are often way more complex in reality, every patient is different
- Modeling errors: discrepancy between mathematical formulation and physical model -> e.g. using dimensionally reduced approaches, like linear dependencies or even constant parameters
- Discretization errors: discrepancy between the continous description and discrete discription of the model
- Solution errors: using iterative approximation methods and rounding errors
- It's basically a butterfly effect
- Optimizing one error source often conflicts with another one -> e.g. handling nonlinearity can cause fatal numerical errors (at least that's what Kroeger said ...)

2.2. Theory of finite elements

2.2.1. Elliptical problems

- Elliptical problems in general
- build up system of PDE's to describe problem
- Using the cylindric domain, different domains

2.2.2. FEM for electrical fields

- special domain
- boundary conditions

2.2.3. FEM for temperature fields

- boundary condition (heat source or sink)

2.2.4. Axial symmetrie

- Using axial symmetrie to simplify computations
- reducing one dimension 3D -> 2D
- significant savings calculations time and complexity
- approach: fourier decomposition in angular direction to reduce dependency on the angular φ
- using static models, only dependency on space
- maybe Torus elements

2.3. This part is about the concrete PDEs itself

2.3.1. FEM in cylindric Coordinates

- Rewrite the equations to cylindric coordinates

Laplace in cartesian coordinates:

$$\nabla^2 := \Delta := \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \tag{1}$$

Laplace in cylindric coordinates:

$$\Delta := \frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} + \frac{1}{r^2}\frac{\partial^2}{\partial \phi^2} + \frac{\partial^2}{\partial z^2} \tag{2}$$

Laplace in polar coordinates:

$$\Delta := \frac{\partial^2}{\partial r^2} + \frac{1}{r}\frac{\partial}{\partial r} + \frac{1}{r^2}\frac{\partial^2}{\partial \phi^2} \tag{3}$$

2.3.2. PDE for Electric potential

Three parts are interesting: - Inner domain
- Fixed Potential of electrodes
- Inner domain
- Outer boundary -> Robin

Constant material parameters:

2.3.3. Inner Domain

- The electric potential of the inner domain is described as :

$$-\nabla \cdot (\sigma(x,y,z,t)\nabla\varphi(x,y,z,t)) = 0 \tag{4}$$

- Elliptical boundary problem
- Assuming constant material parameters: $\nabla\sigma = 0$
- Solution is independent from σ so we can cut it out - Equation becomes Laplaces' equation, phi becomes time independent

$$-\Delta\varphi(x,y,z) = 0 \tag{5}$$

- Using a cylindric domain, we can use cylinder coordinates (see ref Laplace in cylinder)

$$-\Delta\varphi(r,\phi,z) = -\frac{1}{r}\frac{\partial\varphi}{\partial r} - \frac{\partial^2\varphi}{\partial r^2} - \frac{1}{r^2}\frac{\partial^2\varphi}{\partial\phi^2} - \frac{\partial^2\varphi}{\partial z^2} = 0 \tag{6}$$

- Since the domain has axis symmetry, the solution for φ is independant from the angular ϕ
- So equation simplifies to

$$-\frac{1}{r}\frac{\partial\varphi}{\partial r} - \frac{\partial^2\varphi}{\partial r^2} - \frac{\partial^2\varphi}{\partial z^2} = 0 \tag{7}$$

- PDE is now parabolic and no longer elliptic
- We will care about a more complex formulation later

2.3.4. Electrodes

- ϕ = $\pm$ 1

2.3.5. Outer boundary

- For first try, a simplification with natural boundary conditions

$$n \cdot \nabla\varphi = 0 \tag{8}$$

- In cylindrical coordinates

$$TODO \tag{9}$$

2.3.6. PDE for temperature Distribution

Following Kroeger, the temperature distribution is modeled by the heat equation:

$$\partial_t(\rho c T) - \nabla \cdot (\lambda \nabla T) = Q \tag{10}$$

The heat equation is a well known parabolic partial differential equation.

We are assuming ρ and c are constant
ρ = density
c = specific heat capacity
λ = thermal conductivity, which is depending on T
T = T(r,z,t) = temperature
Q = Q(r,z,t) = heat energy

Cylindrical coordinates: see 'Transient Heat Transfer in a Partially Cooled Cylindrical Rod' from Lawrence Agbezuge

$$\rho c \frac{\partial T}{\partial t} - \frac{d\lambda}{dT}\left[\left(\frac{\partial T}{\partial r}\right)^2 + \left(\frac{\partial T}{\partial z}\right)^2\right] - \lambda\left(\frac{\partial^2 T}{\partial r^2} + \frac{1}{r}\frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial z^2}\right) = Q \tag{11}$$

For the first run, we assume *lambda* is also constant too, which greatly reduces the complexity of the problem to the form

$$\rho c \frac{\partial T}{\partial t} - \lambda\left(\frac{\partial^2 T}{\partial r^2} + \frac{1}{r}\frac{\partial T}{\partial r} + \frac{\partial^2 T}{\partial z^2}\right) = Q \tag{12}$$

TODO: explain Q her

$Q = Q_(rf) + Q_{perf}$

3. Applied FEM-Simulation

3.1. Weak solutions

Electric potential / Laplace's equation in cylindrical domain:

$$a_w(u,v) := \int_\Omega (\partial_r u \partial_r v + \partial_z u \partial_z v) r dr dz = \int_\Omega f v r dr dz \tag{13}$$

- $u \in H_r^1(\Omega) \cap \{v|_{\Gamma_0} = 0\}$
- $v \in H_r^1(\Omega) \cap \{v|_{\Gamma_0} = 0\}$

Approximate with linear regression functions
Linear regression functions for reference triangles:

$$\phi_1(\xi,\eta) = 1 - \xi - \eta \tag{14}$$

$$\phi_2(\xi,\eta) = \xi \tag{15}$$

$$\phi_2(\xi,\eta) = \eta \tag{16}$$

Specific PDE for electric potential, inner domain:

$$a_w(u,v) := \int_\Omega (\partial_r u \partial_r v + \partial_z u \partial_z v) r dr dz = 0 \tag{17}$$

3.2. Grid generation / Triangulation

3.3. Get TestData

3.4. Solving the PDEs

3.4.1. numerical challenges

3.4.2. numerical integration

3.4.3. solving the system of equations

3.5. Interpretation of result numbers

3.6. Graphical output

3.7. Optimization

3.8. MatLab vs C++

- Basically the performance advantages of using C++

4. Summary and Outlook

4.1. Project Summary

This is the conclusion part

4.2. strengths and flaws

- why is it good, why is it bad

4.3. State of the current Research

- Research in the simulation of medical therapy methods

4.4. Other FEM projects and software

References

[1] Tim Kröger et. al. *Numerical Simulation of Radio Frequency Ablation with State Dependent Material Parameters in Three Space Dimensions*. Springer, 2006.

[2] Klaus Knothe u. Heribert Wessels. *Finite Elemente, Eine Einführung für Ingenieure, 5. Auflage*. Springer Vieweg, 2017.

[3] Michale Jung u. Ulrich Langer. *Methode der finiten Elemente für Ingenieure, Eine Einführung in die numerischen Grundlagen und Computersimulation, 2. Auflage*. Springer Vieweg, 2013.

[4] Christian G. Sorger. *Generierung von Netzen für Finite Elemente hoher Ordnung in zwei und drei Raumdimensionen*. Technische Universität München, Lehrstuhl für Computation in Engineering, 2012.

[5] Michael McLaughlin. *C++ Succinctly*. Syncfusion Inc., 2012.

[6] Physicists like to think that all you have to do is say 'These are the conditions now what happens next?'. *Richard Feynman*. The Character of Physical Law, 1965.

[7] Not only is the Universe stranger than we think it is stranger than we can think. *Werner Heisenberg*. Across the Frontiers, 1972.

A. Source code Visual C++

Listing 1: For loop to print numbers from 1 to 10

```
// Print numbers from 1 to 10
#include <stdio.h>
int main() {
  int i;
  for (i = 1; i < 11; ++i)
  {
    printf("%d␣", i);
  }
  return 0;
}
```

B. Source code MatLab

TODO