On the Adaptive Time-Stepping in Radio-Frequency Liver Ablation Simulation: Some Preliminary Results

K. Georgiev, N. Kosturski, and Y. Vutov

Institute of Information and Communication Technologies,
Bulgarian Academy of Sciences,

Abstract. Radio-frequency ablation is a low invasive technique for treatment of liver tumors. This work concerns the mathematical modeling and computer simulation of the heat transfer process. The core is solving the time-dependent partial differential equation of parabolic type. Instead of a uniform discretization of the considered time interval, an adaptive time-stepping procedure is applied in an effort to decrease the simulation time. The procedure is based on the local comparison of the Crank Nicholson and backward Euler approximations. Results of some preliminary numerical experiments performed on a selected test problems are presented and discussed.

1 Introduction

The minimally invasive treatment called radio-frequency ablation (RFA), one of several types of ablation therapy, may be the alternative when open surgery of certain cancer types is not a good option. Guided by imaging techniques, the doctor inserts a thin needle through the skin and into the tumor, (see Fig. 1, [19]). High-frequency electrical energy delivered through this needle heats and destroys the tumor. The circuit is closed with a ground pad applied to the patient's skin.

An important advantage of RF current (over previously used low frequency AC or pulses of DC) is that it does not interfere with the muscles and can be used without the need for general anaesthesia.

There is an ongoing research in RF probe design. The right procedure parameters are very important for the successful killing of all of the tumor cells with minimal damage on the non-tumor cells.

Computer simulation on geometry obtained from a magnetic resonance imaging (MRI) scan of the patient is performed. The influence of the position of the ground pad to the ablated volume is of special interest, both from the medical and simulation point of view. Often, in computer simulations reported in the literature e.g. [1, 9, 10, 15], the position of the ground pad is neglected and a simple computational domain with a cubic shape is considered. In [16] the authors check the correctness of the assumption that when the pad is *far* from the probe then zero potential condition can be applied on the whole boundary of

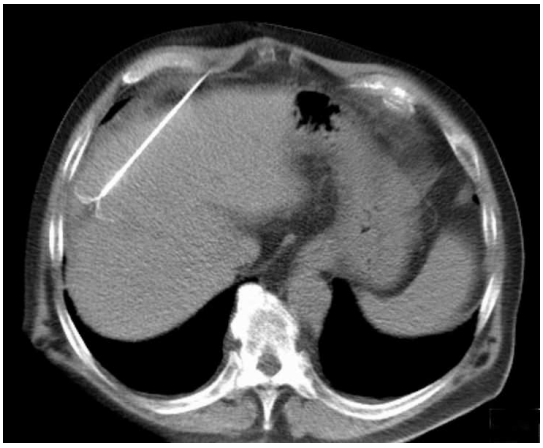

Fig. 1. CT Scan, Showing Radio-Frequency Ablation of a Liver Lesion

the domain and compare the resulting ablated volumes, when ground pads are put in different positions.

In this work, an adaptive time stepping algorithm is applied to the simulation in order to reduce the computational time.

The rest of the paper is organized as follows. In Section 2, the mathematical model is presented along with the space and time discretization schemes. Section 3 describes the adaptive time-stepping algorithm. Section 4 is devoted to the computer simulations and analysis of the results obtained on an IBM Blue Gene/P supercomputer. Finally, some concluding remarks can be found in Section 5.

2 The Model, Space and Time Discretization

As mentioned above, the RFA procedure destroys the unwanted tissue by heating, arising when the energy dissipated by the electric current flowing through a conductor is converted to heat. The considered RF probe consists of a stainless steel needle, insulated with polyurethane. The RFA procedure starts by placing the probe inside the tumor. The surgeon performs this under computed tomography (CT) or ultrasound guidance. The human liver has a complex structure, composed of materials with unique thermal and electrical properties. There are three types of blood vessels with different sizes and flow velocities. Here, a simplified test problem, where the liver consists of homogeneous hepatic tissue and only the large portal vein vessels is considered.
The bio-heat time-dependent partial differential equation [9, 10] is the governing equation describing this process. It can be presented as follows:

$$\rho c \frac{\partial T}{\partial t} = \nabla \cdot k \nabla T + J \cdot E - \alpha \, h_{\mathrm{B}} \, (T - T_{\mathrm{B}}), \tag{1}$$

where the thermal energy arising from the current flow is described by $J \cdot E$ in (1) and $\alpha\, h_{\mathrm{B}}\, (T - T_{\mathrm{B}})$ accounts for the heat loss due to blood perfusion in the capillaries. The heat produced from metabolic functions of the liver is neglected. The initial and boundary conditions which are used in this approach are as follows:

$$T = 37^{\circ}\mathrm{C} \qquad \text{when } t = 0 \text{ at } \Omega, \tag{2a}$$

$$T = 37^{\circ}\mathrm{C} \qquad \text{when } t \geq 0 \text{ at } \partial\Omega, \tag{2b}$$

$$-k\frac{\partial T}{\partial n} = \alpha(T - T_{\mathrm{B}}) \qquad \text{when } t \geq 0 \text{ at } \Gamma_{\mathrm{R}} \tag{2c}$$

The notations which are used in (1) and (2) are given bellow:

- Ω – the entire domain of the model;
- $\partial\Omega$ – the boundary of the domain;
- Γ_r – the boundary of the blood vessel;
- ρ – density [kg/m^3];
- c – specific heat [J/kg K];
- k – thermal conductivity [W/m K];
- J – current density [A/m];
- E – electric field intensity [V/m];
- t – time [s];
- T – temperature [K];
- T_{B} – blood temperature (37°C);
- w_{B} – blood perfusion coefficient$[s^{-1}]$;
- $h_{\mathrm{B}} = \rho_{\mathrm{B}} c_{\mathrm{B}} w_{\mathrm{B}}$ – convective heat transfer coefficient accounting for the blood perfusion in the model;
- α – tissue state coefficient;
- n – the outward-pointing normal vector of the boundary.

The cumulative damage integral $\Psi(t)$ is used as a measure of ablated region [1, 16]:

$$\Psi(t) = \ln\left(\frac{c(0)}{c(t)}\right) = A \int e^{-\frac{\Delta E}{RT(t)}}\, dt, \tag{3}$$

where $c(t)$ is the concentration of living cells, R is the universal gas constant, A is the "frequency" factor for the kinetic expression [s^{-1}], and ΔE is the activation energy for the irreversible damage reaction [J mol^{-1}]. The values used $A = 7.39 \times 10^{39}\mathrm{s}^{-1}$ and $\Delta E = 2.577 \times 10^{5}\mathrm{J\ mol}^{-1}$ are taken from [1]. Tissue damage $\Psi(t) = 4.6$ corresponds to 99% probability of cell death. The value of $\Psi(t) = 1$, corresponding to 63% probability of cell death is significant, because at this point the tissue coagulation first occurs and blood perfusion stops.

The tissue state coefficient α is expressed as

$$\alpha(t) = \begin{cases} e^{-\Psi(t)} \text{ if } \Psi(t) < 1, \\ 0 \text{ if } \Psi(t) \geq 1. \end{cases}$$

In the presented algorithm the bio-heat problem (1) is solved in two steps (see [16] for more details):

1. Finding the heat source $J \cdot E$ using that: (a) $E = -\nabla V$ (V is the electric potential in the computational domain Ω), and (b) $J = \sigma E$, where σ is the electric conductivity [S/m];

2. Finding the temperature T by solving the heat transfer equation (1) using the heat source $J \cdot E$ obtained in the first step.

For the numerical solution of (1) the finite element method in space is used ([12]). *Linear conforming tetrahedral elements* are used in this study. They are directly defined on the elements of the used unstructured mesh (see Fig. 2). An *algebraic multigrid* (AMG) preconditioner is used [6]. The time derivative is discretized via finite differences and the both the *backward Euler* and the *Crank-Nicholson* schemes are used ([13]).

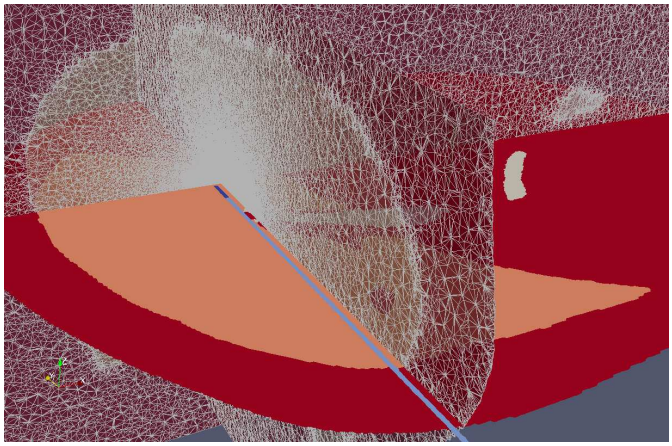

Fig. 2. Inserted RF Probe and the Finite Element Mesh

Let the matrices K and M be the stiffness and mass matrices from the finite element discretization of (1):

$$K = \left[\int_\Omega k \nabla \Phi_i \cdot \nabla \Phi_j d\mathbf{x} \right]_{i,j=1}^{N},$$

$$M = \left[\int_\Omega \rho c \Phi_i \Phi_j d\mathbf{x} \right]_{i,j=1}^{N}.$$

Let us also denote with Ω_B the subdomain of Ω where we account for the blood perfusion (the liver tissue) and with M_B the matrix

$$M_\mathrm{B} = \left[\int_\Omega \delta_\mathrm{B} h_\mathrm{B} \Phi_i \Phi_j d\mathbf{x} \right]_{i,j=1}^{N},$$

where

$$\delta_\mathrm{B}(x) = \begin{cases} \alpha \text{ for } x \in \Omega_\mathrm{B}, \\ 0 \text{ for } x \in \Omega \setminus \Omega_\mathrm{B}. \end{cases}$$

The influence of the Robin boundary conditions given in (2c) and the electric field intensity is presented by:

$$M_{\mathrm{R}} = \left[\int_{\Gamma_{\mathrm{R}}} \alpha \Phi_i \Phi_j d\mathbf{x} \right]_{i,j=1}^{N}, \tag{4}$$

and

$$F = \left[\int_{\Omega} JE \Phi_i \Phi_j d\mathbf{x} \right]_{i,j=1}^{N}, \tag{5}$$

Than, the spatially discretized parabolic equation (1) can be written in matrix form as:

$$M \frac{\partial T}{\partial t} + (K + M_{\mathrm{B}} + M_{\mathrm{R}})T = F + M_{\mathrm{B}} T_{\mathrm{B}} + M_{\mathrm{R}} T_{\mathrm{B}}. \tag{6}$$

3 Adaptive Time-Stepping Algorithm

To ensure accuracy and not waste computational effort, it is important to adapt the time steps to the behavior of the solution.

The time discretization for both backward Euler method and the Crank-Nicolson one can be written in the form

$$\begin{aligned} \left(M + \tau^n \theta (K + M_{\mathrm{B}} + M_{\mathrm{R}})\right) T^{n+1} = {} & \left(M - \tau^n (1-\theta)(K + M_{\mathrm{B}} + M_{\mathrm{R}})\right) T^n \\ & + (\tau^n \theta + \tau^n (1-\theta))(F + M_{\mathrm{B}} T_{\mathrm{B}} + M_{\mathrm{R}} T_{\mathrm{B}}), \end{aligned} \tag{7}$$

where the current (n-th) time-step is denoted with τ^n, the unknown solution at the next time step – with T^{n+1}, and the solution at the current time step – with T^n. If we set the parameter $\theta = 1$, (7) gives a system for the backward Euler discretization. When $\theta = 0.5$ (7) becomes Crank-Nicolson one. The solution of the linear system (7) with $\theta = 1$ and $\theta = 0.5$ gives us T_{BE} and T_{CN} respectively.

A suitable adaptive time-stepping procedure is based on a local comparison of the backward Euler (T_{BE}) and Crank-Nicolson (T_{CN}) approximations for the current timestep, and is controlled by the ratio

$$\eta = \frac{\|T_{\mathrm{CN}} - T_{\mathrm{BE}}\|}{\|T_{\mathrm{BE}}\|}. \tag{8}$$

This approach has a down side, that solving two linear systems is required to obtain T_{BE} and T_{CN}. This is, from the computational point of view, expensive. Nevertheless overall decrease in computational time is expected.

The algorithm below, describing our adaptive time-stepping procedure, is based on the one for adaptive time stepping for processes in spent nuclear fuel repositories [2]. It has several parameters:

1. τ^1 – initial timestep;

2. N_{Adapt} – a parameter showing how often the adaptive time stepping strategy is applied, e.g. $N_{\mathrm{Adapt}} = 1$ shows that the adaptive time stepping is used on each step while $N_{\mathrm{Adapt}} = 3$ – that the adaptive time stepping is performed at every third time step, $N_{\mathrm{Adapt}} = 0$ indicates that all time steps are non-adaptive.
3. $\lambda_{\mathrm{NonAdapt}}$ – a parameter showing whether and by how much the time step is multiplied, in non-adaptive time steps, e.g. $\lambda_{\mathrm{NonAdapt}} = 1$ means that the time step is not changed, while $\lambda_{\mathrm{NonAdapt}} = 1.2$ means that the time step on the current level is multiplied by 1.2 for the next time level.
4. $\varepsilon_{\min}$ and $\varepsilon_{\max}$ are minimal and maximal thresholds for the error estimate η.

Algorithm 1 (Adaptive Time-Stepping Procedure)

1. `for` $k = 1, 2, \ldots$ `until` *the end of time* `do`
2. `if` *CurrentStepIsAdaptive*(N_{Adapt}, k)
2. `then`
3. `do`
4. *compute* T_{BE}, T_{CN} *with* τ^k
5. *compute* η
6. `if` $\eta < \varepsilon_{min}$ `then` $\tau^{k+1} = 2\tau^k$
7. `if` $\eta > \varepsilon_{max}$ `then` $\tau^k = 0.5\tau^k$
8. `while` $\eta > \varepsilon_{max}$ *// if too big error, stay on the same timestep*
9. $T^{k+1} = T_{BE}$
10. `else`
11. *compute* T_{BE} *with* τ^k
12. $T^{k+1} = T_{BE}$
13. $\tau^{k+1} = \tau^k \lambda_{NonAdapt}$
14. `end if`
15. `end for`

The last timestep is always truncated to the time of simulation.

Inner PCG iteration with the BoomerAMG [6] preconditioner, part of the software package HYPRE, is used for the solution of (7). The preconditioner is reconstructed if the number of inner iterations goes above 12. The reconstruction takes place before the solution of the next timestep.

4 Computer Simulations and Analysis of the Output Results

The IBM Blue Gene/P computer, located at the Bulgarian Supercomputing Center, is used for the simulations and numerical experiments with the new adaptive time stepping algorithm. This machine consists of two racks, 2048 Power PC 450 based compute nodes, 8192 processor cores and a total of 4 TB random access memory. Each processor core has a double-precision, dual pipe floating-point core accelerator. Sixteen I/O nodes are connected via fiber optics to a 10 Gbps Ethernet switch.

The material properties which are used in the simulations are taken from [9], and can be seen in Table 1. The blood perfusion coefficient is $w_{\mathrm{B}} = 6.4{\times}10^{-3}\,\mathrm{s}^{-1}$. The applied electrical power is 15 W, and the simulation is done for 7 minutes.

Table 1. Thermal and Electrical Properties of the Materials

Material	ρ (kg/m^3)	c (J/kg K)	k (W/m K)	σ (S/m)
Ni-Ti	6 450	840	18	1×10^8
Stainless steel	21 500	132	71	4×10^8
Liver	1 060	3 600	0.512	0.333
Blood	1 000	4 180	0.543	0.667
Polyurethane	70	1 045	0.026	10^{-5}

We run several test to choose a suitable set of values for the threshold parameters $\varepsilon_{\min}$ and $\varepsilon_{\max}$. As a quantitative criterion of quality of the solution we used two volumes – the volume Vol_1, which is the volume of the tissue, where the cumulative damage integral Ψ is greater than 1, and $Vol_{4.6}$ – the volume of the tissue, where $\Psi > 4.6$. The results of the nonadaptive algorithm with step $\tau = 1$ s were compared with the ones from adaptive runs. Some of the output results obtained on 128 processors on the IBM Blue Gene/P machine are presented in Table 2. Looking at the last four columns in this table one can see that an acceptable variation in the two important volumes less than 3 % occurs when the threshold interval is $[2.5 \times 10^{-4}, 1.25 \times 10^{-3}]$ and this interval is used in the computer simulations. Based on these preliminary tests, a number of runs were done both using 128 and 1024 processors. Uniformly refined mesh was used for the runs on 1024 processors. Some of the output results obtained during the simulations are presented in Table 3 and Table 4. Comparing the total CPU times for 128 and 1024 processors (see the fifth column in both tables) and taking into account that we solve eight times bigger problems on eight times more processors we may conclude that the adaptive time stepping algorithm has excellent scalability. One can see in both tables that the best results with regards

Table 2. Vol_1 and $Vol_{4.6}$ as Functions of the Thresholds in the Adaptive Time-Stepping Algorithm

$\varepsilon_{\min}$	$\varepsilon_{\min}$	Vol_1 [cm^3]	Variation in %	$Vol_{4.6}$ [cm^3]	Variation in
Without adaptive time stepping		22.15	-	15.60	-
5.0×10^{-3}	5.0×10^{-2}	23.72	7.08	16.37	4.90
5.0×10^{-3}	1.0×10^{-2}	23.69	6.99	16.36	4.83
1.0×10^{-3}	5.0×10^{-3}	23.01	3.89	16.06	2.91
5.0×10^{-4}	2.5×10^{-3}	22.94	3.57	16.05	2.84
2.5×10^{-4}	1.25×10^{-3}	22.72	2.57	15.93	2.08

Table 3. Number of Iterations and the CPU Time in the Adaptive Time-Stepping Algorithm in the Case of 128 Processors.

N_{Adapt}	$\lambda_{\text{NonAdapt}}$	No. of inner iterations	No. of outer iterations	CPU time [s]	Vol_1 cm^3	$Vol_{4.6}$ cm^3
0	1.0	2233	420	7608	22.14	15.60
1	1.0	917	102	3968	22.72	15.93
2	1.0	731	104	3137	22.63	15.87
	1.2	535	71	2321	22.87	16.00
	1.3	587	77	2624	22.87	16.02
3	1.0	700	113	3053	22.58	15.83
	1.2	539	76	2329	22.88	16.03
	1.3	592	77	2559	22.81	15.97

Table 4. Number of Iterations and the CPU Time in the Adaptive Time-Stepping Algorithm in the Case of 1024 Processors.

N_{Adapt}	$\lambda_{\text{NonAdapt}}$	No. of inner iterations	No. of outer iterations	CPU time [s]	Vol_1 $[\text{cm}^3]$	$Vol_{4.6}$ $[\text{cm}^3]$
0	1.0	604	420	7259	22.21	15.65
1	1.0	777	101	4234	22.70	15.92
2	1.0	594	101	3488	22.70	15.92
	1.2	478	71	2619	23.01	16.10
	1.3	539	77	2982	22.94	16.07
3	1.0	549	104	3121	22.70	15.93
	1.2	455	76	2530	22.85	16.01
	1.3	514	75	2740	22.94	16.06

to CPU time and number of the inner iterations are obtained when the adaptive strategy is applied at each second time step and meanwhile, at the intermediate time steps τ is multiplied by 1.2. In this case, comparing the total CPU times of the algorithm without the adaptive time-stepping and using this strategy, it is seen that the time of the new algorithm is almost three times shorter.

5 Conclusions

An adaptive time stepping algorithm for simulating the radio-frequency ablation for treatment of liver tumors is presented. The procedure is based on the local comparison of the Crank-Nicholson and the backward Euler approximations. Results of some preliminary numerical experiments performed are presented and discussed. The first experimental results show that the new algorithm is scalable. The tests allowed us to find some suitable parameters and showed the practical usefulness of the developed solver for such kind of computer simulations. One can observe that the computing time is decreased more than three times, the number of outer iterations is decreased from 420 to 71, and the number of inner iteration

decreases form 2233 to 535. This preliminary results are a good motivation for further improving the algorithm and doing more simulations.

Acknowledgments

This research is supported in part by Grants DFNI I01/5 and DCVP-02/1 from the Bulgarian NSF and the Bulgarian National Center for Supercomputing Applications (NCSA) giving access to the IBM Blue Gene/P computer.

References

1. Isaac A. Chang and Uyen D. Nguyen *Thermal modeling of lesion growth with radiofrequency ablation devices* BioMedical Engineering OnLine 2004, 3:27
2. R. Blaheta, P. Byczanski, R. Kohut, J. Stary, *Algorithms for parallel FEM modelling of thermo-mechanical phenomena arising from the disposal of spent nuclear fuel*, in: O. Stephansson, J.B. Hudson, L. Jing (Eds.), Coupled Thermo-Hydro-Mechanical-Chemical Processes in Geosystems, Elsevier, 2004.
3. George Karypis and Vipin Kumar, *Multilevel k-way partitioning scheme for irregular graphs*, Journal of Parallel and DistributedComputing, 48(1), 1998.
4. G. Karypis and V. Kumar, *A coarse-grain parallel multilevel k-way partitioning algorithm*, In Proceedings of the 8th SIAM conference on Parallel Processing for Scientific Computing, 1997.
5. ParMETIS - Parallel Graph Partitioning and Fill-reducing Matrix Ordering `http://glaros.dtc.umn.edu/gkhome/metis/parmetis/overview`
6. V.E. Henson, U.M. Yang, *BoomerAMG: A parallel algebraic multigrid solver and preconditioner*, Applied Numerical Mathematics 41 (1), Elsevier, 2002, 155–177.
7. F. Pellegrini and J. Roman,*Experimental analysis of the dual recursive bipartitioning algorithm for static mapping.* Research Report, LaBRI, Universit e Bordeaux I, August 1996. Available from `http://www.labri.fr/~pelegrin/papers/scotch_expanalysis.ps.gz`
8. *Scotch and PT-Scotch: Software package and libraries for sequential and parallel graph partitioning, static mapping, and sparse matrix block ordering, and sequential mesh and hypergraph partitioning* `http://www.labri.fr/perso/pelegrin/scotch/`
9. S. Tungjitkusolmun, S.T. Staelin, D. Haemmerich, J.Z. Tsai, H. Cao, J.G. Webster, F.T. Lee, D.M. Mahvi, V.R. Vorperian, *Three-dimensional finite-element analyses for radio-frequency hepatic tumor ablation*, IEEE transactions on biomedical engineering 49 (1), 2002, 3–9.
10. S. Tungjitkusolmun, E.J. Woo, H. Cao, J.Z. Tsai, V.R. Vorperian, J.G. Webster, *Thermal-electrical finite element modelling for radio frequency cardiac ablation: Effects of changes in myocardial properties*, Medical and Biological Engineering and Computing 38 (5), 2000. 562–568.
11. O. Axelsson, *Iterative Solution Methods*, Cambridge University Press, 1996.
12. S. Brenner, L. Scott, *The mathematical theory of finite element methods*, Texts in applied mathematics, 15, Springer-Verlag, 1994.
13. E. Hairer, S.P. Norsett, G. Wanner *Solving ordinary differential equations I, II*, Springer Series in Comp. Math., 2000, 2002
14. N. Kosturski, S. Margenov, *Supercomputer Simulation of Radio-Frequency Hepatic Tumor Ablation, AMiTaNS'10 Proceedings*, AIP CP vol. 1301, pp. 486-493.

15. N. Kosturski, S. Margenov, and Y. Vutov, *Comparison of Two Techniques for Radio-frequency Hepatic Tumor Ablation through Numerical Simulation*, AIP Conf. Proc. 1404, 431, [2011]
16. N. Kosturski, S. Margenov, Y. Vutov, *Supercomputer Simulation of Radio-Frequency Hepatic Tumor Ablation, AMiTaNS'12 Proceedings*, AIP CP vol. 1487, pp. 120–126.
17. Lawrence Livermore National Laboratory, *Scalable Linear Solvers Project*, `http://www.llnl.gov/CASC/linear_solvers/`.
18. `http://www.mayoclinic.org/radiofrequency-ablation/`
19. `http://en.wikipedia.org/wiki/File:RFA_CT_Leber_001.jpg`