

Efficient Solver for a Simplified Model of the Multi-Physics Heat Transfer Problem in Radiofrequency Ablation of Hepatic Tumors

Amir Akbari and Dennis D. Giannacopoulos

McGill University, Department of Electrical and Computer Engineering, Montreal, Canada

Abstract—Mathematical modeling and computer simulation of heat transfer in the liver play a key role in prediction of radiofrequency ablation outcome. While multi-physics modeling should be used for accurate simulation of heat transfer in a highly perfused organ like the liver, the computational cost of such a problem could have a negative impact on the simulation time. This paper extends a highly parallel finite element based method to simulate the multi-physics heat transfer problem in the liver efficiently. An MPI-based strong coupling method is developed and tested on cluster nodes. The scalability of this work is assessed by performing simulations on up to 1024 cores.

Index Terms—Finite element method (FEM), heat transfer in the liver, radiofrequency ablation.

I. INTRODUCTION

RADIOFREQUENCY ablation (RFA) is a cancer treatment technique that delivers electrical currents to a tumor in order to damage the cancerous cells by raising the temperature. RFA is the most widely used minimally invasive approach for liver cancer treatment; however, the ablation zone is difficult to control due to various factors such as the cooling effect of the liver vessels. This challenge can result in an imperfect outcome of the treatment potentially leading to a higher local recurrence and a lower survival rate.

In order to address the concerns related to the misprediction of the ablation zone, the use of computational models for RFA has gained interest in the past two decades [1], [2], [3], [4]. Contrast Enhanced Computed Tomography (CECT) allows determination of the vascular anatomy inside the liver and quantification of the liver perfusion. This enables a reliable treatment planning based on simulation models that can be adapted to the physiology of each patient. To simulate RFA therapies and predict the extent of the ablation zone, a multi-physics model is needed. Particularly, the interaction between the liver tissue and the blood vessels inside the liver should be modelled by a non-linear multi-physics simulation.

Multi-physics modeling generally leads to a system of non-linear equations which is usually solved iteratively based on the Newton Raphson (NR) method. At each iteration, a linear system of equations is solved by either directly computing the inverse of the system matrix or iteratively solving a system of algebraic equations starting from an initial guess. Either way, obtaining the solution is computationally expensive. As such, the multi-physics heat transfer problem has often been over-simplified in the literature by either assuming a constant blood temperature or by decoupling of the blood-tissue interaction.

The finite element Gaussian belief propagation (FGaBP) method [5] is a distributed reformulation of the traditional finite-element method (FEM), whose main advantage to FEM is its efficient parallel scalability. This paper presents an

MPI-based multi-physics extension of FGaBP [6] to solve the coupled heat transfer problem in radiofrequency ablation of hepatic tumors. Section II presents the mathematical model of the coupled heat transfer problem. Section III introduces the new parallel multi-physics algorithm. Section IV provides the implementation details, simulation results, and the parallel performance of the algorithm. Finally, Section V delivers concluding remarks.

II. MODELING OF HEAT TRANSFER IN THE LIVER

The most commonly used model for heat transfer in the tissue is the *Pennes* model:

$$\rho c \frac{\partial T}{\partial t} = Q + \nabla \cdot (d \nabla T) + H(T_{bl} - T), \quad (1)$$

where T is the temperature (K), c is the special heat capacity of liver (J/kg/K), ρ is the tissue density (kg/m³), $Q = \sigma |\nabla v|^2$ is the heat source (W/m³), d is the thermal conductivity (W/m/K), H is the convective transfer coefficient (W/m³/K) and T_{bl} is the blood temperature (K). The *Pennes* model assumes the blood temperature is constant, i.e., 310 K. However, this is only true within and close to large vessels [3]. For this reason, Payne *et al.* [4] model the heat transfer problem with two coupled equations:

$$(1-\epsilon)\rho_{ti}c_{ti}\frac{\partial T_{ti}}{\partial t} = (1-\epsilon)Q + (1-\epsilon)\nabla \cdot (d \nabla T_{ti}) + H(T_{bl} - T_{ti}), \quad (2a)$$

$$\epsilon\rho_{bl}c_{bl}\left(\frac{\partial T_{bl}}{\partial t} + \mathbf{v} \cdot \nabla T_{bl}\right) = \epsilon Q + \epsilon \nabla \cdot (d \nabla T_{bl}) - H(T_{bl} - T_{ti}). \quad (2b)$$

In these two equations, subscripts ‘*ti*’ and ‘*bl*’ stand for tissue and blood, respectively. The two additional parameters ϵ and \mathbf{v} are the fraction of blood volume over total volume and blood velocity (m/s), respectively. Note that the blood flow velocity field must be prescribed. Equations (2) are nonlinear due to the linear dependency of thermal conductivity d on temperature. Payne *et al.* [4] solve the coupled blood and tissue equations based on the Picard linearization which is only valid if the non-linearity is mild. In this paper, the Newton Raphson method is used to solve the non-linear coupled equations of (2). The spatial discretization is based on FEM and the backward differentiation formula of order two is employed for time discretization:

$$(1-\epsilon)\rho_{ti}c_{ti}\frac{3T_{ti}^{n+1} - 4T_{ti}^n + T_{ti}^{n-1}}{2\Delta t} = (1-\epsilon)Q + \quad (3a)$$

$$(1-\epsilon)\nabla \cdot (d \nabla T_{ti}^{n+1}) + H(T_{bl}^{n+1} - T_{ti}^{n+1}),$$

$$\epsilon\rho_{bl}c_{bl}\frac{3T_{bl}^{n+1} - 4T_{bl}^n + T_{bl}^{n-1}}{2\Delta t} = \epsilon Q + \epsilon \nabla \cdot (d \nabla T_{bl}^{n+1}) \quad (3b)$$

$$+ H(T_{ti}^{n+1} - T_{bl}^{n+1}) - (\epsilon\rho_{bl}c_{bl})\mathbf{v} \cdot \nabla T_{bl}^{n+1},$$

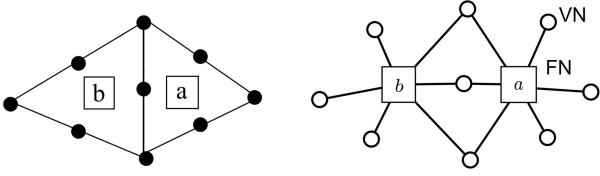


Fig. 1. (Left): Two adjacent elements in the FEM mesh. (Right): The corresponding graph.

where the superscript n denotes the n th time step and Δt is the time step value.

III. ALGORITHM

The belief propagation algorithm was primarily used in probability problems to find marginal distributions of random variables from their joint distribution. If the joint and the marginal distributions are Gaussian, the algorithm is called Gaussian belief propagation (GaBP). The application of GaBP for solving systems of linear equations in parallel was originally proposed by [7] and was extended to FEM applications (FGaBP) by [5], [8]. As shown in Fig. 1, FGaBP turns the FEM mesh into a graph. The graph consists of two types of nodes called variable nodes (VN) and factor nodes (FN) and also non-directional edges that connect each FN to the VNs in its neighborhood. The FEM solution at each VN is considered a Gaussian random variable whose shape is found by two parameters, the marginal mean, μ and the marginal variance, σ . FGaBP finds μ and σ by passing messages between each FN and all the VNs in its neighborhood iteratively. The vector that contains the marginal mean values at convergence is the FEM solution.

In a uni-physics problem, each message contains information about the mean and variance of the VNs. On the other hand, in our multi-physics scenario, the messages should carry information about how the mean and variance of the tissue (blood) temperature depend on the blood (tissue) temperature. By doing so, we can deal with the tissue-blood interaction in the element level. The key property of FGaBP which allows this is that the messages sent by a FN at each FGaBP iteration only depend on the local data, i.e., the values in the neighborhood of the FN. Mathematically, we can write:

$$\begin{cases} m_{ai}^{ti} = f_i(\mathbf{T}_a^{ti}, \mathbf{T}_a^{bl}) \\ m_{ai}^{bl} = g_i(\mathbf{T}_a^{ti}, \mathbf{T}_a^{bl}). \end{cases} \quad (4)$$

Here, m_{ai}^{ti} and m_{ai}^{bl} are messages sent from factor node a to variable node $i \in \mathcal{N}(a)$ for tissue and blood temperature, respectively. The vectors \mathbf{T}_a^{ti} and \mathbf{T}_a^{bl} contain tissue temperature and blood temperature in the neighborhood of factor node a , $\mathcal{N}(a)$. The functions f_i and g_i describe the dependency of the messages on the temperature values in $\mathcal{N}(a)$. It's important to note that f_i (g_i) is a non-linear function of \mathbf{T}_a^{ti} (\mathbf{T}_a^{bl}) because of the temperature dependency of d in (2). At each FGaBP iteration, the non-linear system of (4) can be solved with Newton's method. A local Jacobian

matrix, J_a is constructed based on the functions f_i and g_i as follows:

$$\mathbf{J}_a = \begin{bmatrix} \mathbf{J}_{ti,ti} & \mathbf{J}_{ti,bl} \\ \mathbf{J}_{bl,ti} & \mathbf{J}_{bl,bl} \end{bmatrix}_{2n \times 2n}, \quad (5)$$

where n is the number of nodes per element. The entries of J_a are found based on the derivatives of f_i and g_i with respect to m_{ai}^{ti} and m_{ai}^{bl} . For instance, the sub-matrix $J_{ti,ti}$ is formed as:

$$\mathbf{J}_{ti,ti}(\mathcal{L}(i), \mathcal{L}(j)) = \frac{\partial(m_{ai}^{ti} - f_i)}{\partial m_{aj}^{ti}}, \quad (6)$$

where $1 \leq \mathcal{L}(i), \mathcal{L}(j) \leq n$ are the local indices corresponding to the global variable nodes $i, j \in \mathcal{N}(a)$. These partial derivatives are calculated analytically based on the update rules of FGaBP introduced by [8]. After the local Jacobian is formed, message updates are computed as:

$$\begin{Bmatrix} \{\Delta m_{ai}^{ti}\}_{n \times 1} \\ \{\Delta m_{ai}^{bl}\}_{n \times 1} \end{Bmatrix}_{2n \times 1} = [\mathbf{J}_a^{-1}] \cdot \begin{Bmatrix} \{f_i - m_{ai}^{ti}\}_{n \times 1} \\ \{g_i - m_{ai}^{bl}\}_{n \times 1} \end{Bmatrix}_{2n \times 1}. \quad (7)$$

The above update rule is carried out for each factor node a and then the FGaBP algorithm propagates the updated messages throughout the mesh. After the FGaBP iterations have converged, we proceed to the next Newton iteration and compute the local Jacobians and update messages again. This continues until the local update messages are smaller than a threshold, meaning the cell-wise Newton's method has converged for the current time step.

IV. IMPLEMENTATION AND RESULTS

The computational domain is discretized using hexahedral mesh. Starting by a coarse mesh (and its corresponding graph), mesh refinement is conducted by splitting each hexahedral element into eight smaller elements successively. As proposed by [8], a multi-grid scheme to transfer information between different refinement levels is used to accelerate FGaBP iterations. The multi-grid approach resembles a preconditioner in reducing the number of iterations on the finest level.

An MPI (message passing interface) version of the multi-physics FGaBP method is developed in this paper. The MPI code is built upon the parallel distributed computing in deal.II [9]. The parallel pseudo-code is shown by Algorithm 1. For massively parallel computations, deal.II builds on the p4est [10] library and so does our code for mesh partitioning. Deal.II assigns each part of the partitioned mesh to one MPI process. Each processor stores the cells it owns and also one layer of adjacent cells—called ghost cells—that are owned by other processors. The locally owned cells and ghost cells can be identified by specific indices that deal.II assigns to them. We exploit this feature to define adjacent factor nodes and interface nodes in each sub-domain, and communicate messages between MPI processes. Such communication occurs at the end of each FGaBP iteration. This is shown graphically by Fig. 2. The dashed line is the interface between the two partitions, while the layers shown by grey elements are the ghost layers and are shared between the two partitions.

```

1 for NR iteration  $m = 1, 2, \dots$  do
2   for cell  $a$  in sub-domain  $s$  do
3     for node  $i \in \mathcal{N}(a)$  do
4       Calculate local Jacobian  $J_a$ ;
5       Calculate local residual;
6       Update the messages  $m_{ai}^{ti}$  and  $m_{ai}^{bl}$ ;
7       if  $a$  is an interface cell then
8         if  $i$  is at the interface then
9           Send out  $m_{ai}^{ti}$  and  $m_{ai}^{bl}$  to the
             adjacent sub-domains;
10        end
11      end
12    end
13  end
14  Update global residual;
15  if global tolerance < NR threshold then
16    break;
17  end
18  for FGaBP iteration  $t = 1, 2, \dots$  do
19    for each sub-domain  $s$  in the domain do
20      Load input messages from adjacent
        sub-domains;
21      for cell  $a$  in sub-domain  $s$  do
22        for node  $i \in \mathcal{N}(a)$  do
23          Propagate the messages  $m_{ai}^{ti}$  and
             $m_{ai}^{bl}$  inside  $s$ ;
24          if  $a$  is an interface cell then
25            if  $i$  is at the interface then
26              Send out  $m_{ai}^{ti}$  and  $m_{ai}^{bl}$  to
                the adjacent sub-domain;
27            end
28          end
29          Update  $T_i^{ti}$  and  $T_i^{bl}$ ;
30          Calculate local message residual;
31          Send out local message residual;
32        end
33      end
34      Update global message residual;
35    end
36    if global message residual < FGaBP threshold
      then
37      break;
38    end
39  end
40 end

```

Algorithm 1: Parallel NR with local Jacobian calculation.

Consequently, the messages sent inside elements a and b are communicated between the two processors.

The numerical results of the parallel algorithm are verified against the built-in RFA model in COMSOL Multiphysics software (Fig. 3). More information about the COMSOL model can be found on their website [11]. The model parameters are chosen according to measurement values reported in the literature [4]. A uniform blood velocity field with magnitude between 10-15 cm/s and upward direction (same direction as the cylindrical blood vessel) is employed.

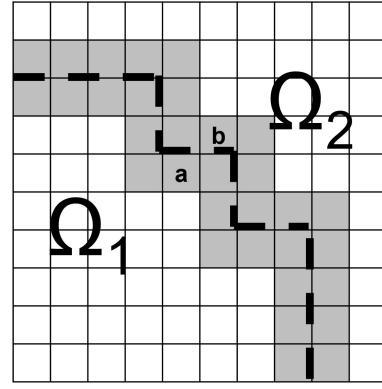


Fig. 2. Mesh partitioning using p4est. The dashed line shows the interface between two sub-domains Ω_1 and Ω_2 .

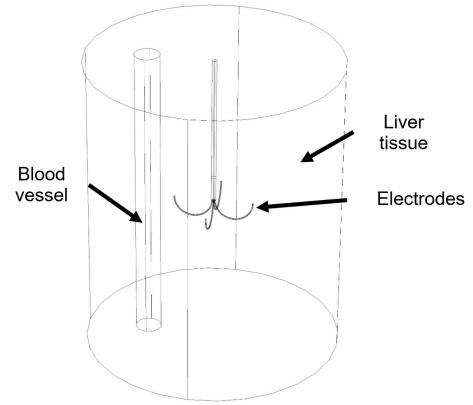


Fig. 3. The built-in COMSOL model of RFA.

Figure 4 depicts two-dimensional thermal maps after 10 minutes of ablation, based on the *Pennes* model (left) and the coupled equations of (2) (right). It shows that assuming a constant blood temperature results in lower tissue temperature in regions that are further apart from large blood vessels. Also, the effect of the blood flow on the thermal map is illustrated by the upward skewness in Fig. 4, right. The tissue temperature from COMSOL and our algorithm at a specific location inside the domain are depicted in Fig. 5. The Root-Mean-Square (RMS) error for 10 minutes of ablation is 0.021 °C.

To assess the parallel efficiency of our method, we perform a strong scaling analysis. The global unstructured mesh is partitioned first. We map each partition to one MPI process (node), and use one thread per process. The algorithm is tested on Compute Canada cluster nodes. Each node contains an Intel Gold 6148 Skylake 2.4 GHz CPU with 186 GB DRAM. Figure 6 shows the speedup for solving the multi-physics system of 200,000 double-precision unknowns. The speed-up degrades from 1 to 2 processes because of the introduction of mesh partitioning and MPI communication. The parallel performance then improves by increasing the number of processes up to 128. After this point, the overhead of communication between the processes and also the assembly of the solution vector dominates the computation time, resulting in the degradation of parallel scalability.

The convergence plot of the parallel NR method is depicted

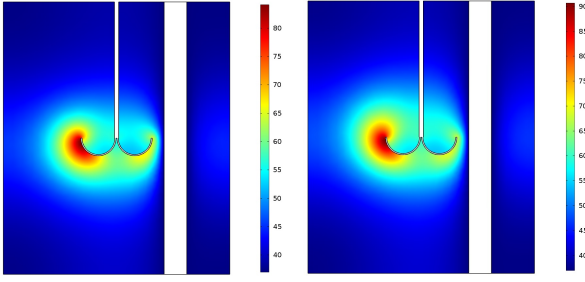


Fig. 4. Thermal maps obtained from simulation of the *Pennes* model (left) and (2) (right).

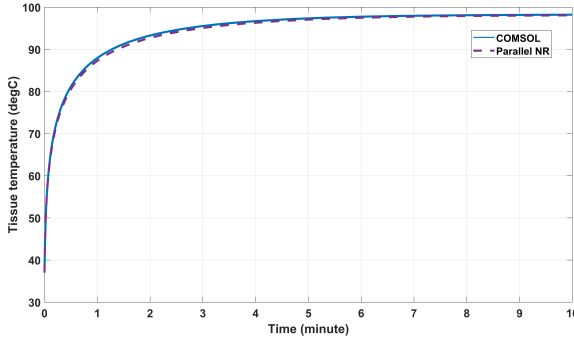


Fig. 5. Comparison between COMSOL and the parallel NR algorithm. Plot shows tissue temperature at a specific point for 10 minutes of ablation.

in Fig. 7. If the initial messages are chosen sufficiently close to the solution, the quadratic convergence can be observed in NR iterations until the convergence criterion is achieved. To ensure the initial guess is close enough to the NR solution, we had to run the algorithm in a Gauss-Seidel manner for a few iterations first.

V. CONCLUSIONS

A FEM-based message passing algorithm is modified and applied to the coupled heat transfer equations that emerge in the modeling of RFA. By forming local Jacobian matrices instead of a global sparse Jacobian, our method provides a highly scalable strong coupling algorithm for solving the non-linear coupled equations on multiple cluster nodes. This

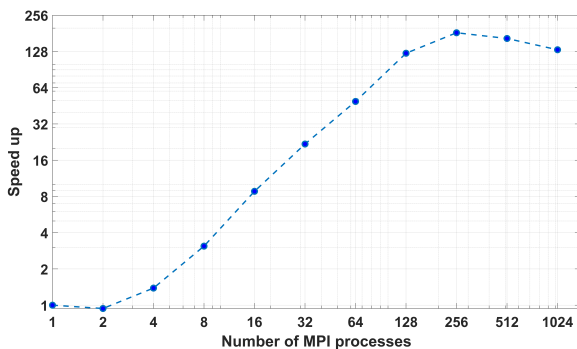


Fig. 6. Strong scaling analysis.

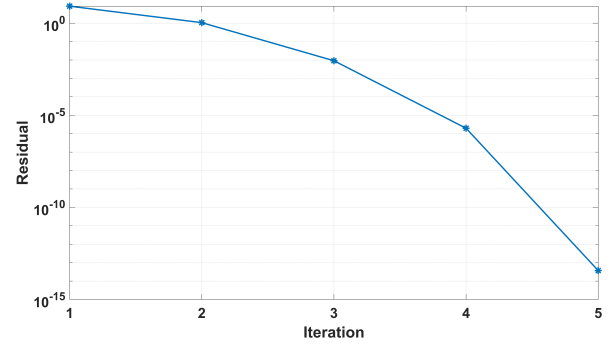


Fig. 7. Convergence plot of the algorithm.

is important in the context of a fast and accurate simulation of RFA treatment for hepatic tumors, considering that the coupled heat transfer problem has often been solved in a decoupled manner because of its high computational cost. The correctness of the algorithm is verified by comparing its results against COMSOL multiphysics software. The parallel scalability of the FGaBP method is retained in the multi-physics algorithm. The parallel implementation shows speedups of more than one hundred times with respect to one MPI processor.

REFERENCES

- [1] M. Bullo, V. D'Ambrosio, F. Dughiero, and M. Guarnieri, "A 3-d cell method formulation for coupled electric and thermal problems," *IEEE Trans. Magn.*, vol. 43, no. 4, pp. 1197–1200, 2007.
- [2] V. D'Ambrosio, P. Di Barba, F. Dughiero, M. Mognaschi, and A. Savini, "Non-invasive thermometry for the thermal ablation of liver tumor: A computational methodology," *International Journal of Applied Electromagnetics and Mechanics*, vol. 25, no. 1-4, pp. 407–412, 2007.
- [3] C. Audigier, T. Mansi, H. Delingette, S. Rapaka, V. Mihalef, D. Carnegie, E. Bector, M. Choti, A. Kamen, N. Ayache *et al.*, "Efficient lattice boltzmann solver for patient-specific radiofrequency ablation of hepatic tumors," *IEEE Transactions on Medical Imaging*, vol. 34, no. 7, pp. 1576–1589, 2015.
- [4] S. Payne, R. Flanagan, M. Pollari, T. Alhonnoro, C. Bost, D. O'Neill, T. Peng, and P. Stiegler, "Image-based multi-scale modelling and validation of radio-frequency ablation in liver tumours," *Phil. Trans. R. Soc. A*, vol. 369, no. 1954, pp. 4233–4254, 2011.
- [5] Y. El-Kurdi, W. J. Gross, and D. Giannacopoulos, "Efficient implementation of gaussian belief propagation solver for large sparse diagonally dominant linear systems," *IEEE Trans. Magn.*, vol. 48, no. 2, pp. 471–474, 2012.
- [6] A. Akbari and D. Giannacopoulos, "An efficient multi-threaded newton-raphson algorithm for strong coupling modeling of multi-physics problems," *Computer Physics Communications*, vol. 258, 2021.
- [7] O. Shental, P. H. Siegel, J. K. Wolf, D. Bickson, and D. Dolev, "Gaussian belief propagation solver for systems of linear equations," in *2008 IEEE International Symposium on Information Theory*. IEEE, 2008, pp. 1863–1867.
- [8] Y. El-Kurdi, W. J. Gross, and D. Giannacopoulos, "Parallel multigrid acceleration for the finite-element gaussian belief propagation algorithm," *IEEE transactions on magnetics*, vol. 50, no. 2, pp. 581–584, 2014.
- [9] W. Bangerth, C. Burstedde, T. Heister, and M. Kronbichler, "Algorithms and data structures for massively parallel generic adaptive finite element codes," *ACM Transactions on Mathematical Software (TOMS)*, vol. 38, no. 2, pp. 1–28, 2012.
- [10] C. Burstedde, L. C. Wilcox, and O. Ghattas, "p4est: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees," *SIAM Journal on Scientific Computing*, vol. 33, no. 3, pp. 1103–1133, 2011.
- [11] C. Bost. Heat transfer in biological tissue with thermal damage analysis. [Online]. Available: <https://www.comsol.com/blogs/heat-transfer-in-biological-tissue-with-thermal-damage-analysis/>