A Novel Optimized Parallelization Strategy to Accelerate Microwave Tomography for Breast Cancer Screening

A. Shahzad, *EMBS Student Member*, M. O'Halloran, *IEEE Member*, M. Glavin, *IEEE Member,* and E. Jones, *IEEE Senior Member*

***Abstract*— Microwave tomography has been proven to successfully reconstruct the dielectric profile of a human breast when used in breast imaging applications, thereby providing an alternative to other imaging modalities. However, the method suffers from high computational requirements which restrict its use in practical imaging systems. This paper presents a novel parallelization strategy to accelerate microwave tomography for reconstruction of the dielectric properties of the human breast. A Time Domain algorithm using this parallelization strategy has been validated and benchmarked against an optimized sequential implementation on a conventional high–end desktop Central Processing Unit (CPU), and a comparison of throughput is presented in this paper. The gain in computational throughput is shown to be significantly higher compared with the sequential implementation, ranging from a factor of 26 to 58, on imaging grid sizes of up to 25 cm square at 1mm resolution.**

I. Introduction

MICROWAVE imaging has been extensively investigated in the area of medical imaging, particularly breast imaging for early stage cancer detection. As a result of this work, a number of clinical prototypes have been developed and recently reported in the literature [1, 2]. Microwave imaging is classified into two categories: radar-based Confocal Microwave Imaging (CMI) techniques e.g. [3], that construct images based on scattered energy from dielectric contrasts in the breast; and microwave tomography [4, 5, 6] that reconstructs the spatial distribution of dielectric properties of the breast tissues using inverse scattering algorithms. Several non-linear inversion algorithms [5, 6] have been developed to reconstruct the dielectric profile of a human breast in microwave tomography. The technique has demonstrated excellent ability to reconstruct the dielectric profile of tissues contained within the breast; however, it comes with a much higher computational cost than CMI. A number of studies have reported the use of clusters of computers connected using a network [5]. These systems suffer performance degradation due to latency and bandwidth limitations of the network interface; however modern GPUs with thousands of parallel computing cores connected directly through the PCIe 3.0 bus can be used to achieve much higher computational efficiency.

A. Shahzad, M. O'Halloran, M. Glavin and E. Jones are with the College of Engineering and Informatics, National University of Ireland Galway, Ireland. (Email: a.shahzad1@nuigalway.ie; martin.ohalloran@nuigalway.ie; martin.glavin@nuigalway.ie; edward.jones@nuigalway.ie)

This paper presents a novel parallelization strategy to accelerate a Time-Domain Inverse Scattering (TDIS) algorithm for reconstruction of the dielectric properties of the human breast. The inversion algorithm is based on the nonlinear conjugate gradient method and is targeted for execution on a massively parallel GPU architecture. An efficient parallelization strategy to compute forward and adjoint solution of electromagnetic scattering is presented. The computation of Fréchet derivative, conjugate directions and Polak-Ribière (PR) constant are accelerated by the parallelization of the method. The results have been verified by comparing with an optimized sequential implementation on an Intel x64 quad-core CPU. The relative improvement in throughput of the parallel implementation compared with the sequential implementation is presented.

The rest of the paper is organized as follows: Section II describes the TDIS algorithm. Section III presents parallelization of the numerical solution of TDIS, and further optimization of the parallel algorithm. Section IV presents the results and discussion. Conclusions and future work are presented in Section V.

II. Time Domain Inverse Scattering

In the TDIS algorithm, an iterative optimization technique is used to minimize the sum of the squared error between measured electromagnetic (EM) signals from the target object itself, and computed EM signals from an estimated numerical model of the target. Consider an array of antennas placed around a breast with unknown dielectric properties as shown in Fig. 1. A set of $M \times N$ measurements is recorded where each of the *M* antennas transmits and the scattered EM signals are recorded on receiving antennas. Another set of $M \times N$ measurements is calculated from an assumed numerical model of the breast, using estimated values of dielectric properties. The cost functional for the minimization of the squared error between these measurements is formulated as:

$$F(\epsilon_r, \sigma) = \int_0^T \sum_{m=1}^{M} \sum_{n=1}^{N} W(t) |E_{m,n}^{meas}(t) - E_{m,n}^{calc}(\epsilon_r, \sigma, t)|^2 dt \quad (1)$$

where $E_{m,n}^{meas}(t)$ is the measured signal at receiving position n corresponding to a transmitted signal from antenna m. The signal $E_{m,n}^{calc}(\epsilon_r, \sigma, t)$ is the forward EM scattering solution computed on a numerical model of the breast with an estimated set of relative permittivity and conductivity values.

978-1-4244-7929-0/14/$26.00 ©2014 IEEE

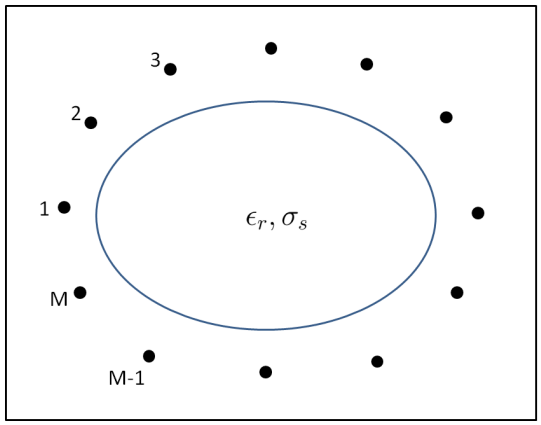

Figure 1. Measurement setup: the breast with unknown dielectric properties surrounded by a circular array of M antennas

The weighting factor $W(t)$ is a non-negative decreasing function of time. T is the measurement interval. The Fréchet derivative of the functional $F(\epsilon_r, \sigma)$ is used to derive gradients with respect to relative permittivity ϵ_r and conductivity σ at each spatial position $r = (x, y)$ in the reconstruction region:

$$G_{\epsilon_r}(r) = 2w_{\epsilon_r} \int_0^T \sum_{m=1}^{M} E^{adj}_{m,r}(\epsilon_r, \sigma, t).\frac{d}{dt}E^{calc}_{m,r}(\epsilon_r, \sigma, t)dt \quad (2)$$

$$G_{\sigma}(r) = 2w_{\sigma} \int_0^T \sum_{m=1}^{M} E^{adj}_{m,r}(\epsilon_r, \sigma, t).E^{calc}_{m,r}(\epsilon_r, \sigma, t)dt \quad (3)$$

where $E^{calc}_{m,r}(\epsilon_r, \sigma, t)$ is the computed EM field at position r in the reconstruction region due to transmitter m, on an estimated model with relative permittivity ϵ_r and conductivity σ. The signal $E^{adj}_{m,r}(\epsilon_r, \sigma, t)$ is the solution to the adjoint field equations, numerically calculated by reverse time propagation of the difference signals from all the receiving antennas back to transmitting antenna m. Additional scaling factors w_{ϵ_r} and w_{σ} are used to compensate for variations in sensitivity of the dielectric parameters. The gradients $G_{\epsilon_r}(r)$ and $G_{\sigma}(r)$ are used with the conjugate gradient method to find the conjugate direction. The complete TDIS algorithm is summarized in Table I.

TABLE I. TDIS ALGORITHM

1.	Start with initial guess, $\epsilon_r^0 = 1$, $\sigma^0 = 0$
2.	For $k = 1, 2, 3, .., n$ to convergence
3.	Compute $E^{calc}_{m,r}(\epsilon_r^k, \sigma^k, t)$
4.	Compute $E^{adj}_{m,r}(\epsilon_r^k, \sigma^k, t)$
5.	Compute Gradients $G^k_{\epsilon_r}$ and G^k_{σ}: eqn. (2)-(3)
6.	Update conjugate directions: β is the PR parameter $d^{k+1}_{\epsilon} = -G^k_{\epsilon_r} + \beta^k_{\epsilon} d^k_{\epsilon}$ $d^{k+1}_{\sigma} = -G^k_{\sigma} + \beta^k_{\sigma} d^k_{\sigma}$
7.	Perform line search: minimize $arg\ \min_{\alpha}\ F(\epsilon_r^k + \alpha d^k_{\epsilon}, \sigma^k + \alpha d^k_{\sigma})$
8.	Update parameters: $\epsilon_r^{k+1} = \epsilon_r^k + \alpha^k d^k_{\epsilon}$ $\sigma^{k+1} = \sigma^k + \alpha^k d^k_{\sigma}$
9.	Test convergence criteria: λ is a predefined threshold $\lvert F(\epsilon_r^k, \sigma^k)\rvert / \lvert F(\epsilon_r^0, \sigma^0)\rvert < \lambda$

III. PARALLELIZATION OF TDIS

The TDIS algorithm requires the computation of forward and adjoint EM scattering solution, Fréchet derivative, conjugate directions, and PR constant at each iteration (Steps 2-6 in Table I). The optimal step size α for updating dielectric properties is determined by a line search in the conjugate direction, using the Golden Section Search (GSS) algorithm of [7].

A. Parallelization

1) Forward and Adjoint Scattering Solution

The forward and adjoint solution to the EM scattering problem is computed here using the Finite Difference Time Domain (FDTD) method [8].

The FDTD method exhibits intrinsic parallelism; each voxel in the FDTD grid is updated using values of neighboring voxels from the previous time step. The adjoint solution is computed using time reversal (TR) in the FDTD method (referred to as TR-FDTD in the following). In the forward FDTD problem, each point in a 2D grid at the N^{th} time step requires neighboring grid values from $(N-1)^{th}$ time step. Similarly, values from the $(N+1)^{th}$ time step are required to update each voxel in TR-FDTD.

2) Computation of Gradients

The computation of gradients $G_{\epsilon_r}(r)$ and $G_{\sigma}(r)$ requires storage of FDTD grid data at each point in the reconstruction region. For each transmitting antenna $m = [1..M]$, a stack of $(X \times Y)$ 2D grids is stored, ; where the stack height is equal to the number of time steps (N), as shown in Fig. 2; X and Y are the dimensions of the FDTD grid used to simulate EM propagation over discretized reconstruction region. Each stack in Fig. 2 is mapped to a 3D computation grid of size $(X \times Y \times N)$. The grid is further divided in blocks of size $(W \times H \times D)$; where parameters Width (W), Height (H) and Depth (D) are chosen to maximize the occupancy of the available resources in the NVIDIA GPU devices used here. Details about the programming model, execution structure and occupancy of the GPUs can be found in [9]. The 3D gradient computation handler multiplies each voxel value of the simulated EM field data with the corresponding voxel of adjoint field data, resulting in a set of M stacks. Each point of stack 1 in the product is summed up with corresponding points in all other stacks. The resultant sum of products is integrated from $[1..N]$ using the trapezoidal method structured to execute on the parallel GPU architecture. A set of differentiated signals is computed prior to application of the gradient computation procedure for permittivity.

3) Conjugate Direction and Line Search

The gradients are used to find the steepest directions, which in this case are opposite to the gradient directions. The

calculation of conjugate directions using the PR method (Step 6 in Table I) and updating the dielectric properties (Step 8 in Table I) involve the same mathematical operations; therefore, these are performed with similar parallel execution structure. Each iteration of the GSS involves evaluation of the cost function at two points, resulting in two FDTD simulations and the computation of cost according to equation (1). The search is terminated according to Step 9 in Table I.

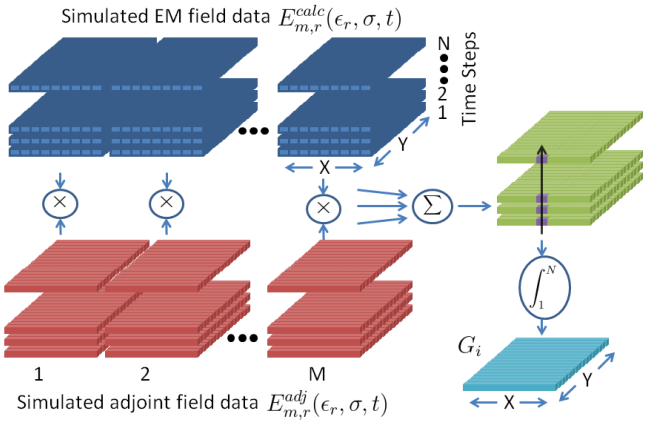

Figure 2. Computation of the gradients of dielectric parameters; *M* is number of transmitting antennas and *N* is number of time steps in the FDTD simulation.

B. *Optimization of the Parallel Algorithm*

Moving from a conventional desktop CPU to a massively parallel GPU architecture for the solution of the FDTD/TR-FDTD, gradient computation, and other arithmetic operations provides a significant gain in the computational throughput in itself. However, a number of further optimizations in the implementation have been introduced to further increase the throughput.

1) Minimizing Data Transfer

Modern GPUs have adequate memory to store the recorded data for 2D image reconstruction, allowing the data to be kept on the GPU to minimize the data transfer between computer's main memory (RAM) and GPU memory, thus saving data transfer time.

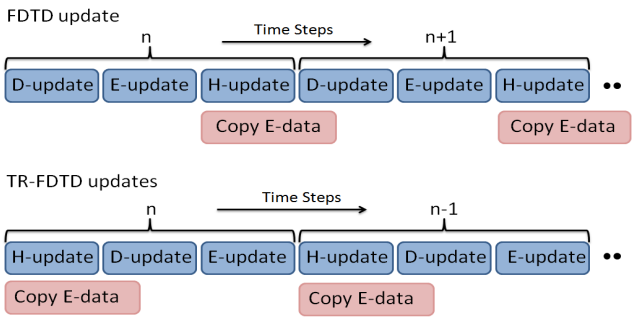

Figure 3. FDTD and TR-FDTD updates with overlapped data transfer

2) Overlapping Data Transfer and Computation

The *E*-field data for each point in the reconstruction region are stored to pre-allocated GPU memory at each time step of the FDTD/TR-FDTD simulation. This GPU-to-GPU memory copy of size $X \times Y \times P$ bytes at each time step is overlapped with FDTD grid update operation as shown in Fig. 3.

3) Coalesced Memory Access

All the loads/stores on a GPU device are carried out through Level 1 (L1) cache and a single load request results in a memory read equal to the number of lines in the cache. If all of the parallel threads read from consecutive memory locations, the number of memory accesses can be minimized.

4) Increasing L1 Cache

The size of L1 cache can be extended on NVIDIA GPUs. This extended configuration of L1 cache further increases the throughput of the parallel model running on the GPU.

IV. RESULTS AND DISCUSSION

The parallel TDIS algorithm was implemented using NVIDIA's CUDA library and the C programming language. The implementation can run on a computer system with NVIDIA's CUDA-ready graphics card. To verify the parallel algorithm and evaluate the gain in comparison with a sequential CPU based implementation, a computer system with an Intel quad core CPU (i7-3770-3.4GHz), NVIDIA's TITAN graphics card, and 16GB of RAM was used. The CPU-based sequential implementation of TDIS was optimized for minimal execution time to produce a fair comparison. Anatomically-realistic MRI-derived numerical breast phantoms from the UWCEM [10] repository have been used to validate the parallel reconstruction algorithm. The numerical model, simulation setup, and other parameters used in this study are the same as those used in [3]. The optimal block size for all 2D kernels (i.e. FDTD, TR-FDTD, conjugate directions, and parameter updating computation) was chosen to be 16×16; the block size for all 3D kernels (i.e. gradient computation: multiply, integrate, differentiate, and sum) was chosen to be $16 \times 16 \times 2$ to achieve maximum occupancy with minimum execution time per kernel.

To compare the computational cost of the parallel implementation of TDIS with a sequential CPU based implementation, the reconstruction of two numerical breast phantoms from the UWCEM repository was performed and the average execution time is given in Table II, for several grid sizes. The cost function is evaluated a number of times during line search, and FDTD simulations are used to compute the forward solution. The average computation on each TDIS iteration during the reconstruction tests required 14 to 20 FDTD/TR-FDTD simulations, and 1 gradient computation. Therefore, a large part of execution time is consumed by FDTD computation. The gain factor in tomography image reconstruction time on a grid size of 250×250 is 29.27 for the parallel implementation before the optimization steps described in section III.B; the gain increased to 58.66 after incorporating the additional optimizations. Table III provides a detailed analysis of the

TABLE II. EXECUTION TIME IN SECONDS FOR SEQUENTIAL CPU-BASED IMPLEMENTATION AND PARALLEL GPU-BASED IMPLEMENTATION BEFORE AND AFTER OPTIMIZATION

Grid Size	CPU (Sec)			GPU (Sec) Before Optim.				GPU (Sec) After Optim.			
	FDTD	Gradient	Avg. time/iter	FDTD	Gradient	Avg. time/iter	Gain	FDTD	Gradient	Avg. time/iter	Gain
50×50	8	1.5	113.5	3.1	0.0012	43.40	2.62	2.2	0.0012	30.80	3.68
100×100	30	6	426	3.52	0.0071	49.29	8.64	2.29	0.0071	32.07	13.28
150×150	68	14	970	4.52	0.054	63.29	15.33	2.59	0.054	36.27	26.74
200×200	127	28	1811	5.88	0.32	82.64	21.91	3.18	0.32	44.84	40.39
250×250	214	75	3140	7.61	0.75	107.29	29.27	3.77	0.75	53.53	58.66

gains provided by the additional optimizations in the parallel implementation.

The first column in the Table III identifies these four configurations of the parallel implementation of FDTD/TR-FDTD:

1. No optimization (standard cache (16KB), and *E*-field data copy to CPU memory.
2. Extended L1 cache (48KB), and *E*-field data copy to CPU memory.
3. Extended L1 cache (48 KB) + *E*-field data copy to GPU memory.
4. Extended L1 cache (48 KB) + E-field data copy to GPU memory + overlapped kernel.

Other columns in Table III show the time taken for each computational operation, while the last column in Table III shows the gain factor relative to Configuration 1 (no optimization), for each of Configurations 2 to 4.

TABLE III. AVERAGE COMPUTATION AND DATA COPY TIME ON A 200×200 GRID FOR EACH FDTD/TR-FDTD UPDATE TIME STEP AGAINST DIFFERENT OPTIMIZED EXECUTION CONFIGURATIONS

Config.	FDTD update kernel			Copy	Total	Gain
	D(μs)	E(μs)	H(μs)	(μs)	(μs)	%age
1	16.27	21.35	24.42	51.49	113.53	0
2	13.95	20.90	23.20	51.49	109.4	3.5%
3	13.95	20.90	23.20	5.05	63.1	42%
4	13.95	20.90	23.20	0	58.05	8.7%

The values provided in Table III are averaged over 1500 time steps. Configuring the shared memory of the GPU as extended L1 cache enhanced the throughput by a factor of 1.04. However, there was a significant increase in the overall performance by keeping the data inside GPU memory. The performance of the parallel TDIS algorithm has been improved by a factor of 1.8 using GPU memory for storage of all data and increased cache configuration. The overall improvement in the gain after all the optimizations is 1.96.

V. CONCLUSIONS AND FUTURE WORK

A parallelization strategy for the microwave tomography of human breast is presented. The algorithm was implemented and verified on NVIDIA's graphics card and compared with an optimized sequential implementation on a desktop CPU. The use of the parallel GPU-based implementation results in an increase in throughput by up to 29 on realistic grid sizes, and up to 56 with additional optimizations. The results suggest that the proposed strategy enables the use of computationally-demanding microwave tomography for practical breast screening systems. Future work will involve investigating a multi-GPU model to enable 3D microwave tomography.

ACKNOWLEDGEMENT

This work is supported by Science Foundation Ireland (grant numbers: 11/SIRG/I2120 and 12/IP/1523)

REFERENCES

[1] J. Bourqui, J. Sill and E. Fear, "A prototype system for measuring microwave frequency reflections from the breast," *International journal of biomedical imaging,* vol. 2012, 2012.

[2] T. Grzegorczyk, P. Meaney, P. Kaufman, R. diFlorio-Alexander and K. Paulsen, "Fast 3-D tomographic microwave imaging for breast cancer detection," *IEEE Transactions on Medical Imaging,* vol. 31, pp. 1584-1592, 2012.

[3] A. Shahzad, M. O'Halloran, E. Jones and M. Glavin, "A preprocessing filter for multistatic microwave breast imaging for enhanced tumour detection," *Progress In Electromagnetics Research B,* vol. 57, pp. 115-126, 2014.

[4] A. E. Bulyshev, S. Y. Semenov, A. E. Souvorov, R. H. Svenson, A. G. Nazorov, Y. E. Sizov and G. P. Tatsis, "Computational modeling of three-dimensional microwave tomography of breast cancer," *IEEE Trans. Biomed. Eng.,* vol. 48, pp. 1053-1056, September 2001.

[5] T. Takenaka, T. Moriyama, K. A. H. Ping and T. Yamasaki, "Microwave breast imaging by the filtered forward-backward time-stepping method," in *Electromagnetic Theory (EMTS), 2010 URSI International Symposium on*, Berlin, 2010.

[6] J. D. Shea, P. Kosmas, S. C. Hagness and B. D. Van Veen, "Three-dimensional microwave imaging of realistic numerical breast phantoms via a multiple-frequency inverse scattering technique," *Medical physics,* vol. 37, no. 8, pp. 4210-4226, Aug 2010.

[7] A. Ruszczynski, "Line Search," in *Nonlinear Optimization*, New Jersey, Princeton University Press, 2006, pp. 213-214.

[8] D. M. Sullivan, Electromagnetic simulation using the FDTD method, Wiley-IEEE Press, 2013.

[9] NVIDIA, "CUDA toolkit documentation," 2013. [Online]. Available: http://docs.nvidia.com/cuda/index.html. [Accessed 12 April 2013].

[10] E. Zastrow, S. K. Davis, M. Lazebnik, F. Kelcz, B. D. V. Veen and S. C. Hagness, "Database of 3D Grid-Based Numerical Breast Phantoms for use in Computational," 2008. [Online]. Available: http://uwcem.ece.wisc.edu/MRIdatabase/. [Accessed 13 12 2013].