Efficient Load Balancing for Parallel Adaptive Finite-Element Electromagnetics With Vector Tetrahedra

Dennis D. Giannacopoulos, Hak Keung Fung, and Baruyr Mirican

Department of Electrical and Computer Engineering, McGill University, Montreal, QC H3A 2A7, Canada

The potential benefits of employing optimal discretization-based (ODB) refinement criteria for vector tetrahedra to achieve load balancing in three-dimensional parallel adaptive finite-element electromagnetic analysis are considered. Specifically, the ability of this class of adaption refinement criteria to resolve effective domain decompositions based on initial discretizations with only relatively few tetrahedra is examined for generalized vector Helmholtz systems. The effectiveness of the new load balancing method is demonstrated with adaptively refined finite-element meshes for benchmark systems.

***Index Terms*—Adaptive systems, electromagnetic analysis, finite-element methods, parallel processing.**

I. Introduction

THE primary benefit of adaptive finite-element methods (AFEMs) is the efficient and accurate computational analysis of continuum problems for only a relatively small fraction of the cost of nonadaptive methods [1]. However, the electromagnetic simulation of some complex problems can still be intractable because of the very large number of degrees of freedom (DOF) necessary for sufficiently accurate solutions. One approach for overcoming this barrier is to combine AFEMs with high-performance computing (HPC) methods such as parallel and distributed simulations [2], [3].

AFEMs evolve nonuniform discretizations. Furthermore, in parallel AFEMs the discretization is partitioned into subdomains, which are then assigned to specific processors. Hence, some subdomains will require more refinement than others as the discretization is adaptively refined. Consequently, an initial discretization that is evenly distributed among processors in terms of workload can lead to a severe workload imbalance as the adaption progresses toward more nonuniform discretizations [2], [3]. This makes the task of balancing computational workload among available processors, in a cost-effective manner, nontrivial.

One disadvantage of many load-balancing schemes is the high interprocessor communication cost associated with assessing the severity of the load imbalance, computing a more balanced work load distribution, and redistributing the improved work load among the available processors at a given iteration of the adaptive process [2]. The purpose of this contribution is to investigate the practical value of an alternative approach to the load-balancing problem for three-dimensional (3-D) vector Helmholtz systems, which requires substantially less communication overall. Specifically, the ability of a class of adaption refinement criteria [1] to efficiently partition a problem into subdomains based on initial discretizations with only relatively few DOF is examined.

Digital Object Identifier 10.1109/TMAG.2006.871434

II. ODB Load Balancing for Vector Tetrahedra

Typically, load-balancing algorithms repeatedly perform remapping or redistribution of DOF to available processors at various stages of the adaption. In the case of remapping, a domain decomposition is recomputed to repartition the entire discretization into subdomains that, ideally, represent equal amounts of computation. Alternatively, DOF can be redistributed among existing subregions in order to balance the processor workload. Relatively sophisticated algorithms have been developed based on both approaches over the past several years [2]. However, communication-to-computation cost ratios that are involved can still be quite high and reduce the overall efficiency of the resulting parallel AFEM process.

An alternative load-balancing approach for parallel AFEMs proposed here, which requires considerably less communication among processors, is to predict an efficient domain decomposition for the entire adaption process based on a single initial discretization containing relatively few DOF. The main advantages of this approach are twofold: it avoids having to solve the load balancing problem repeatedly on large discretizations, and interprocessor communication, for the purpose of load balancing, is required only once.

In this work, ODB refinement criteria [1] are used to assess the relative solution error over an initial finite-element mesh during the early stages of the adaption. Subsequently, subregions are defined once and for all by equally distributing the total estimated solution error over each subregion, as predicted by the refinement criteria. For this approach to work, having approximately equal error in each partition of an initial mesh must result in equal subsequent work for each processor. The validity of this hypothesis is examined in the next section.

A. ODB Refinement Criteria for Vector Tetrahedra

A set of optimization equations for the geometric discretization parameters of vector tetrahedra is developed in [1]. Thus, the relative discretization errors over a finite-element mesh can be estimated in terms of how well these equations are satisfied. ODB refinement criteria are defined implicitly as measures of the residuals of these geometric optimization equations [1]. The newly proposed approach is based on equally distributing these residuals over each partition in a discretization. The essential details of the underlying derivation are provided next for reference.

Consider the first-order vector tetrahedral element (linear edge element), as defined in [4], with vertex positions $(x_l, y_l, z_l), l = 1,2,3,4$. For vector Helmholtz systems where the true solution $\mathbf{E}$ is the stationary point of the complex functional (1)

$$F(\mathbf{E}) = \frac{1}{2}\int_V \left[\frac{1}{\mu_r}(\nabla\times\mathbf{E})\cdot(\nabla\times\mathbf{E}) - k_0^2\varepsilon_r\mathbf{E}\cdot\mathbf{E}\right]dV \tag{1}$$

the optimization equations corresponding to x_l, y_l, and z_l are given by (2), (3), and (4), respectively, evaluated over the tetrahedra that share the vertex in question.

$$\frac{1}{2}E^T\mathbf{V}E - \frac{k_0^2}{2}E^T\mathbf{B}E = 0 \tag{2}$$

$$\frac{1}{2}E^T\mathbf{W}E - \frac{k_0^2}{2}E^T\mathbf{C}E = 0 \tag{3}$$

$$\frac{1}{2}E^T\mathbf{P}E - \frac{k_0^2}{2}E^T\mathbf{D}E = 0. \tag{4}$$

Here E is the field solution vector and k_0 is the free-space wave number of the system. The square matrices $\mathbf{V}$, $\mathbf{W}$, and $\mathbf{P}$ contain the x-, y-, and z-component information, respectively, that corresponds to the first term in (1), for vertex l $(l = 1,2,3,4)$ of a tetrahedron. The entries of these matrices are defined by

$$V_{ij} = \frac{N_{ij}}{(6\mathcal{V})^4}\left[(x_{i_1}-x_{i_2})\frac{\ell_j}{\ell_i}\frac{\partial(x_{i_1}-x_{i_2})}{\partial x_l}\mathcal{V} - \ell_i\ell_j\frac{b_l}{2} + (x_{j_1}-x_{j_2})\frac{\ell_i}{\ell_j}\frac{\partial(x_{j_1}-x_{j_2})}{\partial x_l}\mathcal{V}\right] + \frac{\ell_i\ell_j}{(6\mathcal{V})^4}\frac{\partial N_{ij}}{\partial x_l}\mathcal{V} \tag{5}$$

$$W_{ij} = \frac{N_{ij}}{(6\mathcal{V})^4}\left[(y_{i_1}-y_{i_2})\frac{\ell_j}{\ell_i}\frac{\partial(y_{i_1}-y_{i_2})}{\partial y_l}\mathcal{V} - \ell_i\ell_j\frac{c_l}{2} + (y_{j_1}-y_{j_2})\frac{\ell_i}{\ell_j}\frac{\partial(y_{j_1}-y_{j_2})}{\partial y_l}\mathcal{V}\right] + \frac{\ell_i\ell_j}{(6\mathcal{V})^4}\frac{\partial N_{ij}}{\partial y_l}\mathcal{V} \tag{6}$$

$$P_{ij} = \frac{N_{ij}}{(6\mathcal{V})^4}\left[(z_{i_1}-z_{i_2})\frac{\ell_j}{\ell_i}\frac{\partial(z_{i_1}-z_{i_2})}{\partial z_l}\mathcal{V} - \ell_i\ell_j\frac{d_l}{2} + (z_{j_1}-z_{j_2})\frac{\ell_i}{\ell_j}\frac{\partial(z_{j_1}-z_{j_2})}{\partial z_l}\mathcal{V}\right] + \frac{\ell_i\ell_j}{(6\mathcal{V})^4}\frac{\partial N_{ij}}{\partial z_l}\mathcal{V} \tag{7}$$

where $\mathcal{V}$ is the tetrahedral volume, ℓ_i denotes the length of the ith edge connecting vertices i_1 and i_2 as given in Table I, and b_i, c_i, and d_i are geometric parameters related to the tetrahedron's vertex positions, which can be defined as follows with the subscripts progressing modulo 4:

$$b_i = (-1)^i\begin{vmatrix}1 & y_{i+1} & z_{i+1}\\ 1 & y_{i+2} & z_{i+2}\\ 1 & y_{i-1} & z_{i-1}\end{vmatrix} \tag{8}$$

TABLE I
EDGE DEFINITIONS FOR TETRAHEDRA

Edge i	Vertex i_1	Vertex i_2
1	1	2
2	1	3
3	1	4
4	2	3
5	4	2
6	3	4

TABLE II
EXPLICIT FORMS OF $\partial b_i/\partial y_i$ IN TERMS OF $\mathcal{Z}_{ij}$

i /l	1	2	3	4
1	0	$\mathcal{Z}_{43}$	$\mathcal{Z}_{24}$	$\mathcal{Z}_{32}$
2	$\mathcal{Z}_{34}$	0	$\mathcal{Z}_{41}$	$\mathcal{Z}_{13}$
3	$\mathcal{Z}_{42}$	$\mathcal{Z}_{14}$	0	$\mathcal{Z}_{21}$
4	$\mathcal{Z}_{23}$	$\mathcal{Z}_{31}$	$\mathcal{Z}_{12}$	0

Note: explicit forms of $\partial b_i/\partial z_l$ may be obtained by replacing $\mathcal{Z}_{ij}$ with $\mathcal{Y}_{ji}$; $\partial c_i/\partial x_l$ by replacing $\mathcal{Z}_{ij}$ with $\mathcal{Z}_{ji}$; $\partial c_i/\partial z_l$ by replacing $\mathcal{Z}_{ij}$ with $\mathcal{X}_{ij}$; $\partial d_i/\partial x_l$ by replacing $\mathcal{Z}_{ij}$ with $\mathcal{Y}_{ij}$; $\partial d_i/\partial y_l$ by replacing $\mathcal{Z}_{ij}$ with $\mathcal{X}_{ji}$.

$$c_i = (-1)^{i+1}\begin{vmatrix}1 & x_{i+1} & z_{i+1}\\ 1 & x_{i+2} & z_{i+2}\\ 1 & x_{i-1} & z_{i-1}\end{vmatrix} \tag{9}$$

$$d_i = (-1)^i\begin{vmatrix}1 & x_{i+1} & y_{i+1}\\ 1 & x_{i+2} & y_{i+2}\\ 1 & x_{i-1} & y_{i-1}\end{vmatrix}. \tag{10}$$

Further, N_{ij} is defined in terms of b_i, c_i, and d_i as follows:

$$N_{ij} = (c_{i_1}d_{i_2} - d_{i_1}c_{i_2})(c_{j_1}d_{j_2} - d_{j_1}c_{j_2}) + (d_{i_1}b_{i_2} - b_{i_1}d_{i_2})(d_{j_1}b_{j_2} - b_{j_1}d_{j_2}) + (b_{i_1}c_{i_2} - c_{i_1}b_{i_2})(b_{j_1}c_{j_2} - c_{j_1}b_{j_2}). \tag{11}$$

It may be noted that the partial derivatives of N_{ij} with respect to the element vertex positions, which appear in (5), (6), and (7), can be determined directly from (8)–(11) and Tables I and II, where the quantities $\mathcal{X}_{ij}$, $\mathcal{Y}_{ij}$, $\mathcal{Z}_{ij}$ are defined as follows:

$$\mathcal{X}_{ij} = x_i - x_j \tag{12}$$

$$\mathcal{Y}_{ij} = y_i - y_j \tag{13}$$

$$\mathcal{Z}_{ij} = z_i - z_j. \tag{14}$$

The x-, y-, and z-component information of the second term in the functional (1) are given by the second terms of (2), (3), and (4), respectively, for vertex l $(l = 1,2,3,4)$ of a tetrahedron. The entries of the matrices $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ are defined by

$$B_{ij} = \frac{M_{ij}}{720\mathcal{V}^2}\left[(x_{i_1}-x_{i_2})\frac{\ell_j}{\ell_i}\frac{\partial(x_{i_1}-x_{i_2})}{\partial x_l}\mathcal{V} - \ell_i\ell_j\frac{b_l}{6} + (x_{j_1}-x_{j_2})\frac{\ell_i}{\ell_j}\frac{\partial(x_{j_1}-x_{j_2})}{\partial x_l}\mathcal{V}\right] + \frac{\ell_i\ell_j}{720\mathcal{V}}\frac{\partial M_{ij}}{\partial x_l} \tag{15}$$

TABLE III
EXPLICIT FORMS OF M_{ij} IN TERMS OF f_{ij}

i,j	M_{ij}
1,1	$2\ell_1^2(f_{11}-f_{12}+f_{22})$
1,2	$\ell_1\ell_2(f_{11}-f_{13}-f_{21}+2f_{23})$
1,3	$\ell_1\ell_3(f_{11}-f_{14}-f_{21}+2f_{24})$
1,4	$\ell_1\ell_4(f_{12}-2f_{13}-f_{22}+f_{23})$
1,5	$\ell_1\ell_5(2f_{14}-f_{12}-f_{24}+f_{22})$
1,6	$\ell_1\ell_6(f_{13}-f_{14}-f_{23}+f_{24})$
2,2	$2\ell_2^2(f_{11}-f_{13}+f_{33})$
2,3	$\ell_2\ell_3(f_{11}-f_{14}-f_{13}+2f_{34})$
2,4	$\ell_2\ell_4(2f_{12}-f_{13}-f_{23}+f_{33})$
2,5	$\ell_2\ell_5(f_{14}-f_{12}-f_{34}+f_{23})$
2,6	$\ell_2\ell_6(f_{13}-2f_{14}-f_{33}+f_{34})$
3,3	$2\ell_3^2(f_{11}-f_{14}+f_{44})$
3,4	$\ell_3\ell_4(f_{12}-f_{13}-f_{24}+f_{34})$
3,5	$\ell_3\ell_5(f_{14}-2f_{12}-f_{44}+f_{24})$
3,6	$\ell_3\ell_6(2f_{13}-f_{14}-f_{34}+f_{44})$
4,4	$2\ell_4^2(f_{22}-f_{23}+f_{33})$
4,5	$\ell_4\ell_5(f_{23}-f_{22}-2f_{34}+f_{24})$
4,6	$\ell_4\ell_6(f_{23}-2f_{24}-f_{33}+f_{34})$
5,5	$2\ell_5^2(f_{22}-f_{24}+f_{44})$
5,6	$\ell_5\ell_6(f_{24}-2f_{23}-f_{44}+f_{34})$
6,6	$2\ell_6^2(f_{33}-f_{34}+f_{44})$

Note: $(i,j) = (j,i)$.

$$C_{ij} = \frac{M_{ij}}{720\mathcal{V}^2}\left[(y_{i_1}-y_{i_2})\frac{\ell_j}{\ell_i}\frac{\partial(y_{i_1}-y_{i_2})}{\partial y_l}\mathcal{V} - \ell_i\ell_j\frac{c_l}{6} + (y_{j_1}-y_{j_2})\frac{\ell_i}{\ell_j}\frac{\partial(y_{j_1}-y_{j_2})}{\partial y_l}\mathcal{V}\right] + \frac{\ell_i\ell_j}{720\mathcal{V}}\frac{\partial M_{ij}}{\partial y_l} \tag{16}$$

$$D_{ij} = \frac{M_{ij}}{720\mathcal{V}^2}\left[(z_{i_1}-z_{i_2})\frac{\ell_j}{\ell_i}\frac{\partial(z_{i_1}-z_{i_2})}{\partial z_l}\mathcal{V} - \ell_i\ell_j\frac{d_l}{6} + (z_{j_1}-z_{j_2})\frac{\ell_i}{\ell_j}\frac{\partial(z_{j_1}-z_{j_2})}{\partial z_l}\mathcal{V}\right] + \frac{\ell_i\ell_j}{720\mathcal{V}}\frac{\partial M_{ij}}{\partial z_l} \tag{17}$$

where M_{ij} is given by Table III, in which $f_{ij} = b_ib_j + c_ic_j + d_id_j$. It may be noted that the partial derivatives of (b_mb_n), (c_mc_n), and (d_md_n) with respect to the element vertex positions, which are implicit in (15)–(17), can be determined directly from (8)–(10), and are given for reference in Table IV. Once the optimization equations (2), (3), and (4) for the vertex positions have been evaluated for a given mesh, a weighted sum of their residuals for each element is used to rank the elements for refinement [1]. The practical benefits of using these types of refinement criteria for load balancing in parallel AFEA are examined in the next section.

TABLE IV
EXPLICIT FORMS OF $\partial(b_mb_n)/\partial y_i$ IN TERMS OF b_i AND $\mathcal{Z}_{ij}$

	l			
m,n	1	2	3	4
1,1	0	$2b_1\mathcal{Z}_{43}$	$2b_1\mathcal{Z}_{24}$	$2b_1\mathcal{Z}_{32}$
1,2	$b_1\mathcal{Z}_{34}$	$b_2\mathcal{Z}_{43}$	$b_2\mathcal{Z}_{24}+b_1\mathcal{Z}_{41}$	$b_2\mathcal{Z}_{32}+b_1\mathcal{Z}_{13}$
1,3	$b_1\mathcal{Z}_{42}$	$b_3\mathcal{Z}_{43}+b_1\mathcal{Z}_{14}$	$b_3\mathcal{Z}_{24}$	$b_3\mathcal{Z}_{32}+b_1\mathcal{Z}_{21}$
1,4	$b_1\mathcal{Z}_{23}$	$b_4\mathcal{Z}_{43}+b_1\mathcal{Z}_{31}$	$b_4\mathcal{Z}_{24}+b_1\mathcal{Z}_{12}$	$b_4\mathcal{Z}_{32}$
2,2	$2b_2\mathcal{Z}_{34}$	0	$2b_2\mathcal{Z}_{41}$	$2b_2\mathcal{Z}_{13}$
2,3	$b_3\mathcal{Z}_{34}+b_2\mathcal{Z}_{42}$	$b_2\mathcal{Z}_{14}$	$b_3\mathcal{Z}_{41}$	$b_3\mathcal{Z}_{13}+b_2\mathcal{Z}_{21}$
2,4	$b_4\mathcal{Z}_{34}+b_2\mathcal{Z}_{23}$	$b_2\mathcal{Z}_{31}$	$b_4\mathcal{Z}_{41}+b_2\mathcal{Z}_{12}$	$b_4\mathcal{Z}_{13}$
3,3	$2b_3\mathcal{Z}_{42}$	$2b_3\mathcal{Z}_{14}$	0	$2b_3\mathcal{Z}_{21}$
3,4	$b_4\mathcal{Z}_{42}+b_3\mathcal{Z}_{23}$	$b_4\mathcal{Z}_{14}+b_3\mathcal{Z}_{31}$	$b_3\mathcal{Z}_{12}$	$b_4\mathcal{Z}_{21}$
4,4	$2b_4\mathcal{Z}_{23}$	$2b_4\mathcal{Z}_{31}$	$2b_4\mathcal{Z}_{12}$	0

Note: $(m,n) = (n,m)$ explicit forms of $\partial(b_mb_n)/\partial z_l$ may be obtained by replacing $\mathcal{Z}_{ij}$ with $\mathcal{Y}_{ji}$; $\partial(c_mc_n)/\partial x_l$ by replacing b_i with c_i and $\mathcal{Z}_{ij}$ with $\mathcal{Z}_{ji}$; $\partial(c_mc_n)/\partial z_l$ by replacing b_i with c_i and $\mathcal{Z}_{ij}$ with $\mathcal{X}_{ij}$; $\partial(d_md_n)/\partial x_l$ by replacing b_i with d_i and $\mathcal{Z}_{ij}$ with $\mathcal{Y}_{ij}$; $\partial(d_md_n)/\partial y_l$ by replacing b_i with d_i $\mathcal{Z}_{ij}$ with $\mathcal{X}_{ji}$.

B. Mesh Partitioning Algorithms for Load Balancing

To focus ideas, three mesh partitioning strategies for load balancing are considered. The first approach, *Uniform/NonAdaptive*, is based on partitioning an initial mesh into subregions, containing equal numbers of DOF. The second, *Uniform/Adaptive*, is based on uniformly partitioning the discretization into subregions containing equal numbers of DOF after several adaptive refinements of the initial mesh. Finally, the new approach uses ODB refinement criteria to partition the same mesh as the second method into subregions of equal estimated solution error.

III. RESULTS

The effectiveness of the new load-balancing approach is examined using a 3-D vector Helmholtz system defined by Fig. 1(a). The system consists of an air-filled rectangular cavity with perfectly conducting walls, excited at the TE_{101} resonant frequency. One-half of the cavity was initially discretized using 40 tetrahedra based on subdividing each of the eight hexahedra shown into five tetrahedra—the symmetry plane is defined by $z = 0$. The full cavity has dimensions 2 cm, 1 cm, and 2 cm in the x, y, and z directions, respectively, and was analyzed using first-order vector tetrahedra [4]. The relative performance of the three mesh partitioning strategies described above was assessed based on h-refinements.

Fig. 1 illustrates some sample partitioned meshes. The initial 40-element mesh (eight hexahedra) used for all three strategies is shown in Fig. 1(a), as well as the four partitions (labeled A, B, C, and D) used for the *Uniform/Nonadaptive* method. The partitions for the *Uniform/Adaptive* strategy and the new approach are shown in Fig. 1(b) and (c), respectively. Note that both of these meshes are identical, by definition, but the subdomains assigned to the various processors are quite different. For the *Uniform/Adaptive* strategy, the partitioning is according to the criterion of equal numbers of DOF per subdomain; for the new approach, equal error per subdomain was the objective, as

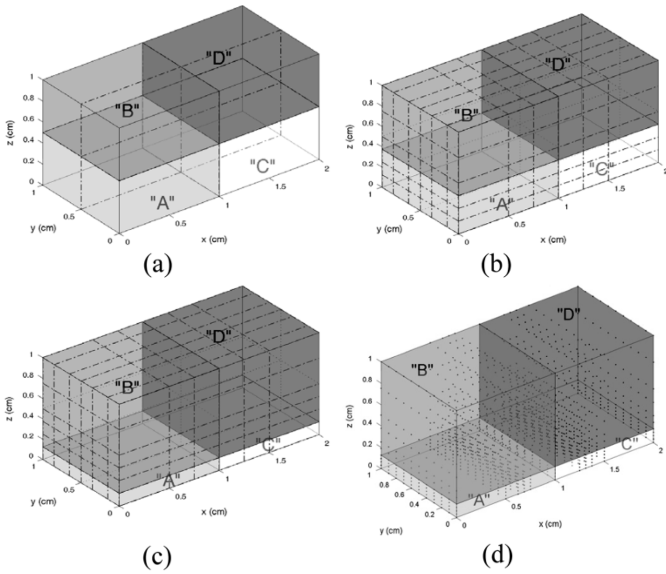

Fig. 1. Partitions for test system. (a) *Uniform/Nonadaptive*, with 40 tetrahedra. (b) *Uniform/Adaptive*, with 1080 tetrahedra. (c) New approach, with the same mesh as in Fig 1(b), and a different subregions partitioning. (d) Sample vertices for new approach, with 5000 tetrahedra.

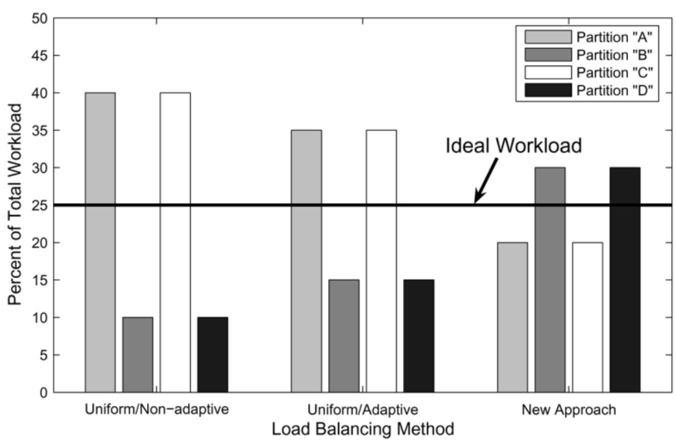

Fig. 2. Vector Helmholtz test system load-balancing results.

explained in the previous section. A sample mesh in terms of vertex distribution for the new approach is shown in Fig. 1(d).

Performance results for the load-balancing studies are presented in Fig. 2. The new approach achieves an improved workload balance compared to the other two approaches (the maximum imbalance among processors is 10% for the new approach versus 20% for the *Uniform/Adaptive* and 30% for the *Uniform/Nonadaptive* approaches). Note that for each approach, the adaptive process was terminated when the global solution percent error was below 0.1%, and the results reported in Fig. 2 are based on these terminal meshes.

A comparison of the solution times of the various approaches considered above with that of a basic single machine serial solution is given in Table V. The results are based on a symmetric multiprocessor with four identical processors (Sun v880) [3]. The load balancing achieved by the new approach results in improved solution times relative to the other methods (26% and 53% relative to the *Uniform/Adaptive* and *Uniform/Nonadaptive* approaches, respectively).

TABLE V
SPEED-UP RELATIVE TO SINGLE MACHINE SERIAL SOLUTION TIME

Method	Speedup
Uniform/Non-Adaptive	1.976
Uniform/Adaptive	2.415
New Approach	3.043

IV. CONCLUSION

A new load-balancing approach for parallel AFEM for 3-D vector Helmholtz systems that uses OBD refinement has been proposed and evaluated. The results for the electromagnetic benchmark system investigated demonstrate that this new approach is able to achieve effective domain partitions based on a single initial discretization with only relatively few DOF. Compared with existing techniques, the new approach requires significantly less interprocessor communication since load balancing and mesh partitioning are done once, and only once, and the adaptive mesh refinement then proceeds without additional need for load balancing. These successful findings suggest further development and experimental studies are merited. For example, the focus of future work should include applying the method to more complex models and examples. This will be essential for the application of the new approach to increasingly realistic systems such as transient problems involving eddy currents and spatial motions.

ACKNOWLEDGMENT

This work was supported in part by the Natural Sciences and Engineering Research Council of Canada.

REFERENCES

[1] D. Giannacopoulos and S. McFee, "Optimal discretization based adaptive finite element analysis for electromagnetics with vector tetrahedra," *IEEE Trans. Magn.*, vol. 37, no. 5, pp. 3503–3506, Sep. 2001.

[2] C. J. Liao and Y. C. Chung, "Tree-based parallel load-balancing methods for solution-adaptive finite element graphs on distributed memory multicomputers," *IEEE Trans. Parallel Distrib. Syst.*, vol. 10, no. 4, pp. 360–370, Apr. 1999.

[3] S. McFee, Q. Wu, M. Dorica, and D. Giannacopoulos, "Parallel and distributed processing for h-p adaptive finite element analysis: A comparison of simulated and empirical studies," *IEEE Trans. Magn.*, vol. 40, no. 2, pp. 928–933, Mar. 2004.

[4] J. Jin, *The Finite Element Method in Electromagnetics*. New York: Wiley, 1993.

Manuscript received June 17, 2005 (e-mail: dennis.giannacopoulos@mcgill.ca).