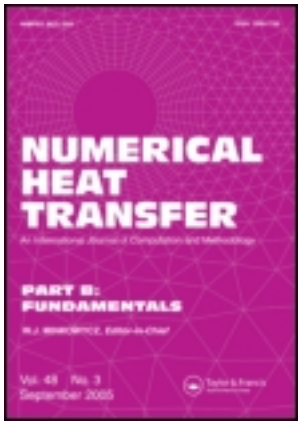

Numerical Heat Transfer, Part B: Fundamentals: An International Journal of Computation and Methodology
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/unhb20

BOUNDARY-ELEMENT ANALYSIS OF 3-D DIFFUSION PROBLEMS USING A PARALLEL DOMAIN DECOMPOSITION METHOD

M. S. Ingber [a] , C. C. Schmidt [b] , J. A. Tanski [b] & J. Phillips [b]
[a] Department of Mechanical Engineering, University of New Mexico, Albuquerque, New Mexico, USA
[b] Engineering Science and Applications Division, Los Alamos National Laboratory, Los Alamos, New Mexico, USA
Published online: 02 Feb 2011.

To cite this article: M. S. Ingber , C. C. Schmidt , J. A. Tanski & J. Phillips (2003) BOUNDARY-ELEMENT ANALYSIS OF 3-D DIFFUSION PROBLEMS USING A PARALLEL DOMAIN DECOMPOSITION METHOD, Numerical Heat Transfer, Part B: Fundamentals: An International Journal of Computation and Methodology, 44:2, 145-164, DOI: 10.1080/713836344

To link to this article: http://dx.doi.org/10.1080/713836344

Numerical Heat Transfer, Part B, 44: 145–164, 2003

ISSN: 1040-7790 print/1521-0626 online
DOI: 10.1080/10407790390200143

Taylor & Francis
Taylor & Francis Group

BOUNDARY-ELEMENT ANALYSIS OF 3-D DIFFUSION PROBLEMS USING A PARALLEL DOMAIN DECOMPOSITION METHOD

M. S. Ingber
Department of Mechanical Engineering, University of New Mexico, Albuquerque, New Mexico, USA

C. C. Schmidt, J. A. Tanski, and J. Phillips
Engineering Science and Applications Division, Los Alamos National Laboratory, Los Alamos, New Mexico, USA

A parallel domain decomposition method is developed for the solution of three-dimensional diffusion problems. Within each subdomain, time is discretized using the generalized trapezoidal rule. The resulting modified Helmholtz equation is solved using the particular-solution boundary-element method. Interfacial conditions between subdomains are satisfied using a Schwarz Neumann–Neumann iteration scheme. Except for the first time step, when zero initial flux is assumed on all interfacial boundaries, the initial estimates for the interfacial flux are given from the converged solutions from the previous time step. This significantly reduces the number of iterations required to meet the convergence criterion.

1. INTRODUCTION

The boundary-element method (BEM) has become an attractive alternative technique to domain methods such as the finite-difference and finite-element methods for solving partial differential equations. The method has proven to be particularly conducive for a wide class of linear elliptic boundary-value problems because of the inherent reduction in the dimensionality of the problem. In fact, in many cases, the BEM requires only the discretization of the boundary of the domain. This can be particularly advantageous in a design setting where the creation of a discretized representation of the solid model often represents the major portion of the overall effort. This is also advantageous in situations in which the problem geometry changes with time. Unfortunately, some of the advantages of the BEM are lost for nonelliptic problems. Nevertheless, the BEM has been successfully applied to a variety of parabolic and hyperbolic partial differential equations.

Received 7 October 2002; accepted 28 February 2003.
This work was supported by Los Alamos National Laboratory for the U.S. Department of Energy under contract DE-AC04-94AL85000.
Address correspondence to M. S. Ingber, University of New Mexico, Department of Mechanical Engineering, Albuquerque, NM 87131, USA. E-mail: ingber@me.unm.edu

NOMENCLATURE

$[A_{ij}]$, $[B_{ij}]$, $[C_{ij}]$, $[D_{ij}]$	boundary-element coefficient matrices	α	thermal diffusivity
		α_i^n	ith radial basis function coefficient at time step n
c_p	specific heat at constant pressure	β_i	ith radial basis function spread coefficient
g	internal heat source		
h	heat transfer coefficient	γ	relaxation parameter for the Neumann-Neumann iteration
k	thermal conductivity		
$\bar{q}$	prescribed heat flux	Γ	boundary of the computation domain
$q_m^{i,n}$	imposed interfacial heat flux for the ith subdomain, nth time step, and mth iteration step		
		ϵ	error criterion for the Schwarz Neumann-Neumann iteration
R	interfacial residual for the Schwarz Neumann-Neumann iteration	θ	generalized trapezoidal method parameter
t	time	ρ	density
Δt	time step	ψ_i	ith radial basis function
u	temperature	$[\psi_{mi}]$	matrix to determine radial basis function coefficient
u_∞	ambient temperature		
$\bar{u}$	prescribed temperature	ϕ	particular solution associated with the radial basis function, ψ
v	transformed temperature		
v_g^n	general solution at time step n	$[\tilde{\Phi}_{ji}]$	coefficient matrix to determine the particular solution
v_p^n	particular solution at time step n		
$v_m^{i,n}$	interfacial transformed temperature for the ith subdomain, nth time step, and mth iteration step	Ω	compuational domain

Several boundary-element formulations have previously been developed to solve parabolic partial differential equations. The first boundary-element formulation used to solve the diffusion equation was proposed by Rizzo and Shippy [21]. Rizzo and Shippy eliminated the time derivative using the Laplace transform to reduce the problem to an elliptic form and then employed a direct BEM. The inverse transform was performed numerically at a discrete set of points. However, as the number of points increased, the matrix associated with the inversion process became ill-conditioned and led to numerical instabilities [14]. Improvements to the inversion process have been developed to mitigate some of the problems caused by this ill-conditioning [16].

The next genre of BEMs applied to the diffusion equation used a time-dependent Green's function [2, 14, 25]. These methods, designated as total Green's function methods, retained the full dimensionality of the problem. The associated integrals possessed integrands that approached Dirac delta functions as the time step was reduced. At the approach of delta-function behavior, solution errors actually increased with decreasing time step because of increased quadrature errors [3]. Taigbenu and Liggett [24] developed a method which performed the time integrations analytically to avoid the problem associated with these large quadrature errors for small time steps.

The third genre of BEMs used discretization in time [2, 10, 26] transforming the diffusion equation into a sequence of either Poisson or nonhomogeneous modified Helmholtz equations. In particular, these methods avoided the cumbersome inverse transforms required by the Laplace transform methods and eliminated the time

integrations required by the total Green's function methods. However, a disadvantage of the early BEMs based on discretization in time is that they required performing domain integrations. For some BEMs, domain integrations can account for the vast majority of the overall CPU effort [1]. Several techniques have recently been developed for eliminating the domain integrals associated with boundary-element methods [8, 11, 17, 27].

Boundary-element methods using discretization in time to solve diffusion problems can become memory- and CPU-intensive for large problems. One method of reducing the CPU requirements is to employ a domain decomposition method. Domain decomposition methods have been extensively studied previously for finite-element methods [7, 13, 22]. Domain decomposition for boundary-element methods have been studied for potential problems by Kamiya et al. [12], Davies and Mushtaq [4], and Mai-Duy et al. [15], for viscous fluid flow problems by Power and Mingo [19], and for thermal convection problems by Power and Mingo [20]. However, all of these previous BEM investigations have dealt with two-dimensional stationary problems.

In the current research, a domain decomposition method is developed for the solution of three-dimensional diffusion problems using the boundary-element method and discretization in time. Interfacial boundary-conditions are updated using a Schwarz Neumann–Neumann method [12]. The boundary-element formulation is discussed in Section 2. The domain decomposition method along with the parallel implementation is discussed in Section 3. The accuracy and parallel efficiency of the domain decomposition BEM is demonstrated through three benchmark problems in Section 4, and conclusions are presented in Section 5.

2. BOUNDARY-ELEMENT FORMULATION

The model diffusion equation is given by the heat conduction equation with constant conductivity k, density ρ, and specific heat c_p in a three-dimensional domain Ω bounded by the surface Γ. The governing equation is given by

$$\rho c_p \frac{\partial u}{\partial t} + g = k\nabla^2 u \tag{1}$$

where u is the temperature and g represents internal heat generation. The boundary conditions for this problem can be any combination of the following types:

$$\begin{aligned} &\text{Prescribed temperature:} \qquad u = \bar{u} \\ &\text{Prescribed flux:} \qquad q = -\bar{q} \\ &\text{Convection:} \qquad q = -h(u - u_\infty) \end{aligned} \tag{2}$$

where q is the heat flux defined by $q = k\,\partial u/\partial n$, h is the heat transfer coefficient, and u_∞ is the ambient temperature. The initial condition is given by

$$u(\mathbf{x}, 0) = f(\mathbf{x}) \tag{3}$$

The method of discretization in time is employed, whereby the spatial variables are discretized using the boundary-element method and time is discretized using the

finite-difference method. In particular, the generalized trapezoidal method (θ method) is used to approximate the time derivative as

$$\frac{1}{\alpha}\frac{u^n - u^{n-1}}{\Delta t} = \theta\left(\nabla^2 u^n - \frac{g^n}{k}\right) + (1-\theta)\left(\nabla^2 u^{n-1} - \frac{g^{n-1}}{k}\right) \tag{4}$$

where the diffusivity, α, is defined by $\alpha = k/\rho c_p$, Δt is the time step, $u^n = u(\mathbf{x}, n\Delta t)$, and $g^n = g(\mathbf{x}, n\Delta t)$. Defining the transformed temperature, v^n, as

$$v^n = u^n - \frac{\theta - 1}{\theta} u^{n-1} \tag{5}$$

the above equation can be rearranged as

$$\nabla^2 v^n = \frac{v^n}{\theta\alpha\,\Delta t} - \frac{1}{\theta^2\alpha\,\Delta t} u^{n-1} + \frac{(1-\theta)}{\theta k} g^{n-1} + \frac{g^n}{k} \tag{6}$$

In the current BEM approach, the right-hand side of (6) can be considered as a "generalized" forcing function. Following standard formulation procedures for the BEM, v^n may be represented in terms of a boundary integral as

$$\eta(\mathbf{x})v^n(\mathbf{x}) = \int_\Gamma [G'(\mathbf{x},\boldsymbol{\xi})v^n(\boldsymbol{\xi}) - G(\mathbf{x},\boldsymbol{\xi})v^{n'}(\boldsymbol{\xi})]\, d\Gamma(\boldsymbol{\xi})$$
$$+ \int_\Omega \left[\frac{v^n(\boldsymbol{\xi})}{\theta\alpha\Delta t} - \frac{1}{\theta^2\alpha\,\Delta t} u^{n-1}(\boldsymbol{\xi}) + \frac{(1-\theta)}{\theta k} g^{n-1}(\boldsymbol{\xi}) + \frac{g^n(\boldsymbol{\xi})}{k}\right] G(\mathbf{x},\boldsymbol{\xi})\, d\Omega(\boldsymbol{\xi}) \tag{7}$$

where $G(\mathbf{x},\boldsymbol{\xi})$ is the Green's function given by $G(\mathbf{x},\boldsymbol{\xi}) = 1/|\,\mathbf{x} - \boldsymbol{\xi}\,|$, the prime denotes the derivative in the direction of the outward normal to the boundary at the point $\boldsymbol{\xi}$, and the coefficient $\eta(\mathbf{x})$ can be determined from the integral

$$\eta(\mathbf{x}) = \int_\Gamma G'(\mathbf{x},\boldsymbol{\xi}) d\Gamma(\boldsymbol{\xi}) \tag{8}$$

The major difficulty with this approach is that the domain integral contains the unknown function $v^n(\boldsymbol{\xi})$. Although v^n can be discretized in the interior using nodal basis functions, the evaluation of the domain integral is usually very time-consuming. In order to eliminate the domain integral, the term inside the square brackets in (7) is approximated as

$$f(\mathbf{r}) = \frac{v^n}{\theta\alpha\,\Delta t} - \frac{1}{\theta^2\alpha\,\Delta t} u^{n-1} + \frac{(1-\theta)}{\theta k} g^{n-1} + \frac{g^n}{k} \approx \sum_{i=1}^{M} \alpha_i^n \psi_i(\mathbf{r}) \tag{9}$$

The functions ψ_i have the form

$$\psi_i(\mathbf{r}) = \psi\left(\frac{|\,\mathbf{r} - \mathbf{r}_i\,|}{\beta_i}\right) \tag{10}$$

where $\mathbf{r}_i$ is a fixed position vector and β_i is a constant. The ψ_i's are called radial basis functions.

There is a wide variety of possibilities for the functional form of the radial basis functions and the parameters β_i, and $\mathbf{r}_i$. In the current implementation, the radial

basis functions are chosen from the family of multiquadric functions [9]. The particular form of the radial basis functions is given by

$$\psi_i(\mathbf{r}) = (1 + r^2)^{-1/2} \tag{11}$$

where $r = | \mathbf{r} - \mathbf{r}_i | / \beta_i$. The $\mathbf{r}_i$'s are chosen to be fairly evenly distributed points within the interior of the domain, Ω, and the β_i's are chosen to be the same order of magnitude as the distance of the closest neighbors to $\mathbf{r}_i$. These choices for the parameters lead to a good interior approximation for $f(\mathbf{r})$ which can be determined from a well-conditioned set of linear equations for the α_i's [27]. A good reference on the theory of radial basis functions, which discusses in some detail the selection of the appropriate parameters, is provided by Powell [18].

Collocating (9) at each of the M interior points $\mathbf{r}_i$ yields a set of equations which can be written in matrix form as

$$[\psi_{mi}]\{\alpha_i^n\} = \frac{\bar{v}_m^n}{\alpha\theta\,\Delta t} - \frac{u_m^{n-1}}{\alpha\theta^2\,\Delta t} + \frac{g_m^n}{k} + \frac{(1-\theta)}{\theta k} g_m^{n-1} \tag{12}$$

where $\bar{v}_m^n$, u_m^{n-1}, and g_m^n represent the values of v, u, and g, respectively, at the mth interior node $\mathbf{r}_m$ at the time step indicated by the superscript. The tilde is used to distinguish v in the interior from v on the boundary in the subsequent discussion. This equation can be rearranged into the following form:

$$\{\bar{v}_m^n\} = \alpha\theta\,\Delta t[\psi_{mi}]\{\alpha_i^n\} + \frac{1}{\theta}\{u_m^{n-1}\} - \frac{\alpha\theta\,\Delta t}{k}\left[\{g_m^n\} + \frac{(1-\theta)}{\theta}\{g_m^{n-1}\}\right] \tag{13}$$

Associated with the radial basis function, $\psi(r)$, is a particular solution $\phi(r)$ such that

$$\nabla^2\phi = \frac{1}{r^2}\frac{d}{dr}\left(r^2\frac{d\phi}{dr}\right) = \psi(r) \tag{14}$$

The particular solution is given by

$$\phi(r) = \frac{1}{2}\left[(1+r^2)^{1/2} + \frac{\ln(r + (1+r)^{1/2})}{r}\right] \tag{15}$$

Hence, an approximate particular solution of (6), $v_p^n(r)$, at the nth time step is given by

$$v_p^n(r) = \sum_{i=1}^{M} \alpha_i^n \phi_i(r) \tag{16}$$

Using the change of variables

$$v_g^n = v^n - v_p^n \tag{17}$$

where v_g^n is the general solution at time step n, the governing equation for v_g^n is given by

$$\nabla^2 v_g^n = 0 \qquad \mathbf{x} \in \Omega \tag{18}$$

The boundary conditions associated with the transformed problem can be determined from (2). The boundary integral equation associated with the transformed problem no longer contains a domain integral term as in (7). That is,

$$\eta(\mathbf{x})v_g^n(\mathbf{x}) = \int_\Gamma [G'(\mathbf{x},\boldsymbol{\xi})v_g^n(\boldsymbol{\xi}) - G(\mathbf{x},\boldsymbol{\xi})v_g^{n'}(\boldsymbol{\xi})]\, d\Gamma(\xi) \tag{19}$$

Collocating (19) at each of the boundary-element nodes yields a set of n linear equations, which can be written in matrix form as

$$[A_{ij}]\{v_{g_j}^n\} = [B_{ij}]\{q_{g_j}^n\} \tag{20}$$

where $\{v_{g_j}^n\}$ represents the value of the general solution, v_g, at the nth time step at the jth boundary element node and $\{q_{g_j}^n\}$ represents the corresponding flux (normal derivative). The boundary conditions for the transformed problem are related to the particular solution, which in turn can be related to the coefficients α_i of the interior approximation by

$$\{v_{p_j}^n\} = [\Phi_{ji}]\{\alpha_i^n\} \qquad \{q_{p_j}^n\} = [\Phi'_{ji}]\{\alpha_i^n\} \tag{21}$$

where Φ_{ji} represents the value of ϕ_i at the jth boundary-element node and Φ'_{ji} represents the value of the normal derivative of ϕ_i at the jth boundary-element node. Inserting (17) and (21) into (20) yields

$$[A_{ij}]\{v_j^n\} - [A_{ij}][\Phi_{jk}]\{\alpha_k^n\} = [B_{ij}]\{q_j^n\} - [B_{ij}][\Phi'_{jk}]\{\alpha_k^n\} \tag{22}$$

The α_i^n's have been introduced into (22) as additional unknowns, since the interior approximation given by (9) is indeterminate. Therefore, it is necessary to introduce additional equations in order to close the algebraic system. This is accomplished by collocating (19) at the M interior points $\mathbf{r}_i$ associated with the radial basis functions. The resulting equations can be written as

$$\{\tilde{v}_{g_i}^n\} + [C_{ij}]\{v_{g_j}^n\} = [D_{ij}]\{q_{g_j}^n\} \tag{23}$$

where $\tilde{v}_{g_i}^n$ represents the general solution, v_g, at the interior point $\mathbf{r}_i$. In the interior, the transformed variable is given by

$$\tilde{v}_{g_i}^n = \tilde{v}_i^n - \tilde{v}_{p_i}^n \tag{24}$$

where $\tilde{v}_i^n$ and $\tilde{v}_{p_i}^n$ represent the values of v and associated particular solution v_p, respectively, at $\mathbf{r}_i$ at the nth time step. Again, $\tilde{v}_{p_i}^n$ can be related to the α_i^n's through the equation

$$[\tilde{\Phi}_{ji}]\{\alpha_i^n\} = \{\tilde{v}_{p_j}^n\} \tag{25}$$

where $\tilde{\Phi}_{ji}$ represents the value of ϕ_i at $\mathbf{r}_j$. Inserting (13), (24), and (25) into (23) yields

$$\alpha\theta\,\Delta t[\psi_{mi}]\{\alpha_i^n\} - [\tilde{\Phi}_{mi}]\{\alpha_i^n\} + \frac{1}{\theta}\{u_m^{n-1}\} - \frac{\alpha\theta\,\Delta t}{k}\left[\{g_m^n\} + \frac{1-\theta}{\theta}\{g_m^{n-1}\}\right]$$
$$+ [C_{mj}]\{v_j^n\} - [C_{mj}][\Phi_{jk}]\{\alpha_k^n\} = [D_{mj}]\{q_j^n\} - [D_{mj}][\Phi'_{jk}]\{\alpha_k^n\} \tag{26}$$

The system of equations given by (22) and (26) can now be solved for $\{v_j^n\}$ and $\{\alpha_k^n\}$ by standard techniques to trace the time history of the boundary-element solution. As a practical matter, all matrices can be evaluated outside the time loop. Further, an LU decomposition can also be performed outside the time loop. Hence, the only operations performed within the time loop are matrix-vector multiplications and back-substitution.

3. PARALLEL DOMAIN DECOMPOSITION METHOD

In the parallel domain decomposition method, the three-dimensional domain Ω is subdivided into multiple subdomains. At each time step, neither the potential u nor flux q is known along the interfaces between the subdomains. In general, there are three iterative approaches for determining the interfacial potential and flux [12]. In Uzawa's method, the potential is assumed on all interfacial boundaries and the potential is iteratively updated, in part, by considering the resulting mismatch in flux across the interfaces. In the Schwarz Dirichlet–Neumann method, the potential is assumed on one side of an interface while the flux is assumed on the other side of the interface. Here, the interfacial potential is updated by considering the resulting mismatch in interfacial potential. In the Schwarz Neumann–Neumann method, the flux is assumed on all interfacial boundaries and the flux is again updated, in part, by considering the mismatch in interfacial potential.

The Schwarz Neumann–Neumann method is chosen in the current research for a technical reason. The boundary-element method implementation uses double nodes along subdomain edges and triple nodes at corners [23]. In order to avoid having to perform off-functional node collocation, Dirichlet conditions cannot be specified at two or more coincident nodes along an interfacial edge or corner. In order to avoid this situation, the Schwarz Neumann–Neumann method is chosen since all interfacial boundary conditions specify an assumed value for the flux.

Consider an interface between subdomain i and subdomain j. At the initial time step, all interfacial fluxes are assumed to be zero. That is,

$$q_0^{i,0} = 0 \qquad q_0^{j,0} = 0 \tag{27}$$

where the subscript indicates the iteration number, the first superscript indicates the domain number, and the second superscript indicates the time step number. After the first time step, the initial interfacial fluxes are given by the converged values from the previous time step. That is,

$$q_0^{i,n} = q^{i,n-1} \qquad q_0^{j,n} = q^{j,n-1} \tag{28}$$

where $q^{i,n-1}$ represents the converged value of the flux on the i-subdomain side of the interface at the $(n-1)$ time step.

The assumed interfacial conditions are modified using

$$q_m^{i,n} = q_{m-1}^{i,n} + \gamma(v_{m-1}^{i,n} - v_{m-1}^{j,n}) \tag{29}$$

$$q_m^{j,n} = q_m^{i,n} \tag{30}$$

where γ is a positive relaxation parameter. The chosen value of γ will affect the convergence rate of the Neumann–Neumann iteration as discussed in the next

section. The optimal value of γ is problem-dependent and, in general, must be determined through numerical experiment [5, 12].

The parallel implementation of the algorithm is based on the Single Program, Multiple Data (SPMD) paradigm with Message Passing Interface (MPI) for parallel communication. Each subdomain is assigned to a processor. Although theoretically, any processor could accommodate more than one subdomain, in the current implementation, only one subdomain is allowed per processor.

Each boundary-element analysis performed for each subdomain on separate processors is independent and does not require any communication. After solving for the unknown boundary data and interior unknowns (the α_i's used in the interior approximation), a MPI_allgather of these unknowns is performed. The allgather is a global MPI operation in which all unknowns are collected on all processors. This is a potentially time-consuming communications operation but was required since one-sided communications such as "put"s and "get"s as contained in the MPI2 standard are not generally available. The one-sided communications would obviate the need for the global allgather.

With all boundary and interior data now available on all processors, a convergence criterion can be set. The convergence criterion used in this research is given in terms of the maximum interfacial residual, R, defined by

$$R = \max |v_m^{k,n} - v_m^{l,n}| < \epsilon \tag{31}$$

where the maximum is taken over all interfacial nodes and ϵ is a prescribed tolerance. If the criterion is met, the converged values of the flux are used as the initial estimates for the next time step. If not, the interfacial fluxes are updated for the next iteration according to (29) and (30).

The only other communication required besides the allgather discussed above is an MPI_Allreduce at the end of each iteration step. Each interface is considered either a master or a slave surface. The convergence criterion is checked on all master surfaces, and the processor sets the convergence flag to 1 if all its master surfaces satisfy the convergence criterion or 0 otherwise. The MPI_Allreduce takes the minimum of all convergence flags from each processor and stores the result in the convergence flag variable on each processor. At the end of the MPI_Allreduce, all processors have either a 1 or a 0 stored in the convergence flag variable, and hence, each processor will go to either the next iteration step or the next time step.

4. BENCHMARK DEMONSTRATION PROBLEMS

All benchmark problems were run on the Los Lobos Linux cluster at the University of New Mexico, which is comprised of dual-processor 733-MHz Pentium III processors. Timings are based on the wall clock speed measured in seconds.

4.1. Problem 1

The governing equation for the first benchmark problem is given by

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} + g(x, y, z, t)$$

to be solved in the domain $0 \leq x, y, z \leq 1$. The forcing function is given by

$$g(x, y, z, t) = 3 \sin x \sin y \sin z (3 \sin t + \cos t)$$

The initial and boundary conditions are appropriate to the solution

$$u = 1 + 3 \sin x \sin y \sin z \sin t$$

The dependent variable u is prescribed on the face $x = 1$ and the flux q is prescribed on all other faces. This problem is of interest since the boundary conditions and forcing function are time-dependent.

The unit cube is subdivided into 8, 27, or 64 cubic subdomains of equal size. Within each subdomain, the mesh included 294 boundary-element nodes, 54 boundary elements, and 125 centers of the radial basis functions. The system of equations given by (22) and (26) has rank 419 (= 294 + 125) for this particular discretization. A value of $\theta = 0.5$ is chosen for the time discretization (Crank–Nicolson scheme), since this results in a second-order-accurate truncation error [6].

Results at the interior point $x = 0.75$, $y = 0.75$, $z = 0.75$ for the 8- and 27-subdomain discretizations are shown in Figure 1. Both discretizations provide excellent results, with the largest pointwise errors of the order of 0.5%.

The convergence characteristics of the Schwarz Neumann–Neumann iteration method are shown in Figures 2–4. In Figure 2, the minimum and maximum number of iterations to satisfy the convergence criterion (31) is shown for $\epsilon = 1.0E - 4$ and $\epsilon = 1.0E - 6$. These results were generated using the 8-subdomain discretization, and

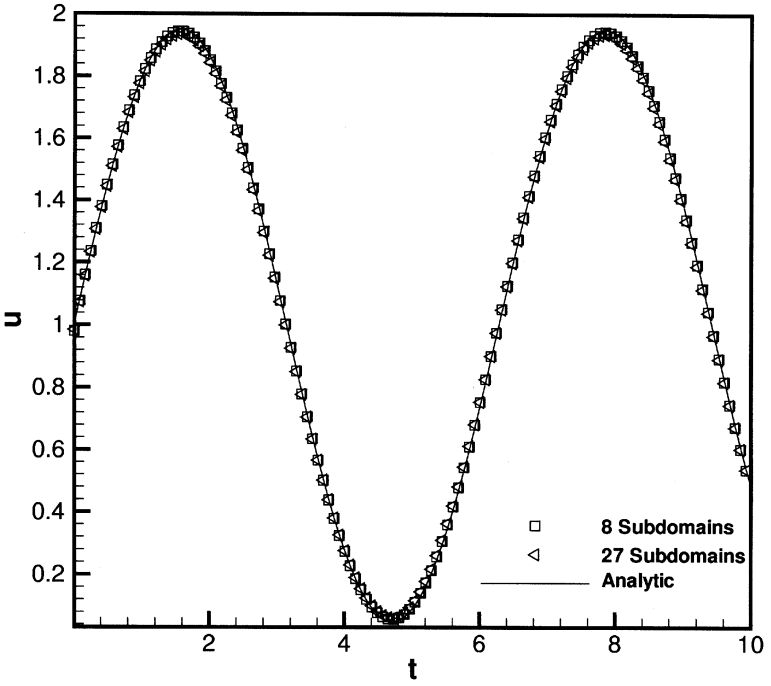

Figure 1. Comparison at the point $x = y = z = 0.75$ between the 8-subdomain BEM, 27-subdomain BEM, and analytic results for benchmark problem 1.

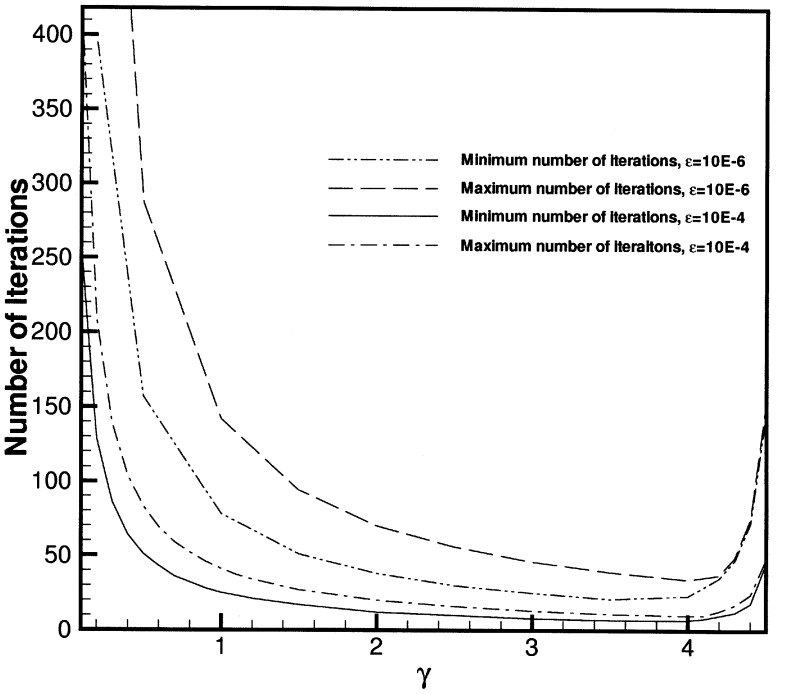

Figure 2. Minimum and maximum number of iterations for the subdomain method to converge as a function of γ for benchmark problem 1 using 8 subdomains.

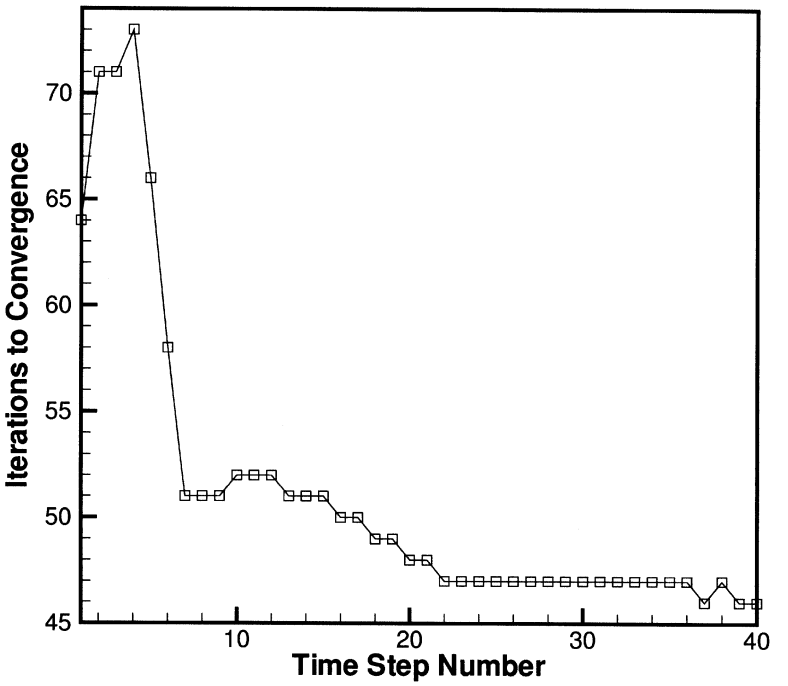

Figure 3. The number of iterations for convergence as a function of the time step for benchmark problem 1 using 8 subdomains and $\gamma = 4.0$.

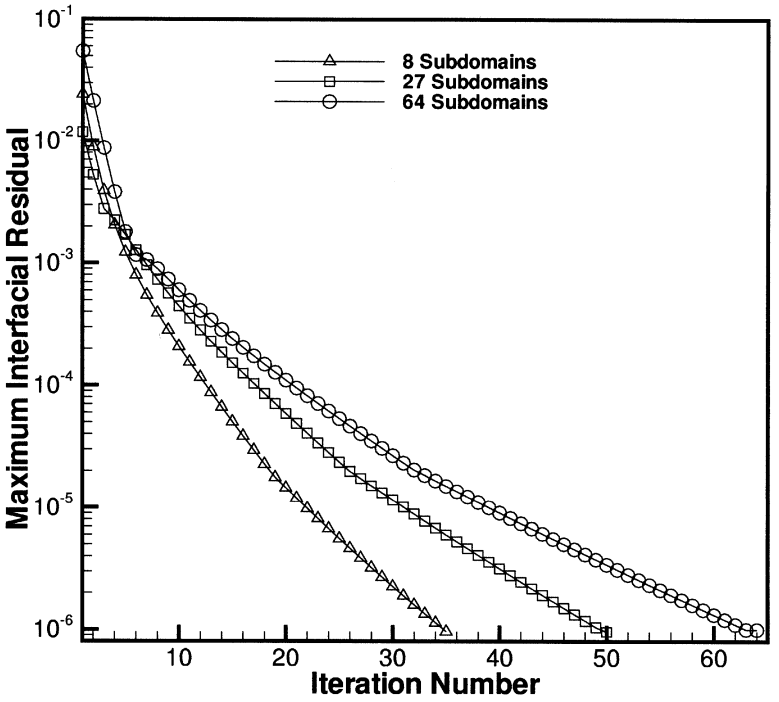

Figure 4. The maximum interfacial residual as a function of the iteration number for the initial time step in benchmark problem 1 for three different discretizations and $\gamma = 4.0$.

the minimum and maximum number of iterations were taken over the first 40 time steps. It is interesting to observe that overrelaxation of the iteration, that is, choosing $\gamma \approx 4.0$, minimizes the overall iteration count. In fact, choosing γ between 2 and 4 provides relatively fast convergence. However, for γ greater than approximately 4.5, the Schwarz Neumann–Neumann method ceases to converge at all.

For the first time step, all interfacial fluxes are initially set to zero. After the first time step, the initial interfacial flux at the new time step is chosen to be the converged interfacial flux from the previous time step. With better initial estimates, the iteration count at a given time step can be reduced. This reduction in the number of iterations is shown in Figure 3, again for the 8-subdomain case and $\epsilon = 1E - 6$.

A plot of the maximum interfacial residual, R, for the 8-, 27-, and 64-subdomain cases as a function of the iteration number at the initial time step is shown in Figure 4 for $\gamma = 4.0$. It is seen in the figure that the iteration count increases sublinearly with increasing number of subdomains. Also as seen in the figure, the reduction in maximum interfacial residual decreases monotonically with each iteration step.

The three discretizations used in benchmark problem 1 provide an excellent means of measuring the scaled parallel efficiency of the BEM subdomain method, since each subdomain contains exactly the same number of nodes and elements. Therefore, each processor uses approximately the same amount of memory and performs approximately the same number of operations. The total wall clock time and the wall clock time per iteration for the 8-, 27-, and 64-subdomain cases are shown in Table 1. Again, as previously seen for the initial time step (Figure 4), the

Table 1. Wall clock times for problem 1 using $\gamma = 3.0$ and $\epsilon = 1.0E-6$

Number of subdomains	Total number of iterations	Total wall clock time (s)	Wall clock time per iteration (s)
8	31,246	1,475	0.0472
27	44,057	1,765	0.0401
64	56,054	2,346	0.0419

total number of iterations over 1,000 time steps increases sublinearly with increasing number of subdomains. However, the wall clock time per iteration is essentially constant, which shows an almost perfect scaled parallel efficiency. This indicates that the communication costs are insignificant in comparison to the computational costs.

4.2. Problem 2

This problem considers a cubic region $0 \leq x, y, z \leq 1.0$ which is initially at 1°C. The sides with $x = 0$, $y = 0$, and $z = 0$ are insulated. For time $t > 0$, the remaining three sides are placed in a convective environment with far-field temperature equal to zero. That is, convection boundary conditions are specified on three sides of the cube. The thermal diffusivity and heat transfer coefficients are given by $\alpha = 1.0$ and $h = 1.0$.

As in the previous benchmark problem, the unit cube is subdivided into cubic subdomains of equal size containing 294 boundary-element nodes, 54 boundary elements, and 125 centers of the radial basis functions. For this benchmark, cases including 8, 27, 64, 125, and 216 subdomains were considered. Again, the Crank–Nicolson scheme is chosen for time discretization in which $\theta = 0.5$. Comparisons between the subdomain BEM results and analytic results are shown at two points for the 8-subdomain case in Figure 5. As seen in the figure, the subdomain BEM provides reliable results.

Results of convergence studies for the Schwarz Neumann–Neumann iteration method similar to those performed for the first benchmark problem are shown in Figures 6–8. For this problem, the minimum and maximum number of iterations required for convergence ($\epsilon = 1.0E-6$) for both the 27 and 64 subdomain cases is shown in Figure 6. It is interesting to note, that as in the previous benchmark problem, overrelaxation of the parameter γ is required to minimize the iteration count. It is also interesting to note that $\gamma \approx 5.7$ resulted in the minimum number of iterations for both the 27- and 64-subdomain cases. The largest γ shown in the figure for the 64-subdomain case is $\gamma = 5.8$, because $\gamma = 5.9$ resulted in a nonconvergent iteration to determine the interfacial potentials. However, as seen in the figure, for $3 < \gamma < 5.8$, the iteration counts are fairly close to minimal values.

The number of iterations until convergence as a function of the time step number is shown in Figure 7 for the case $\gamma = 4.0$ and 64 subdomains. As seen in the figure, the number of iterations at the 50th time step is about third of the initial number of time steps. This is a greater percent reduction compared to the previous benchmark problem. The difference in the two problems is that the second problem is approaching a steady state, whereas no steady state exists for the first problem. Presumably, the number of iterations would go down to 1 as steady state is approached, since the initial interfacial flux estimate would be the correct steady-state value.

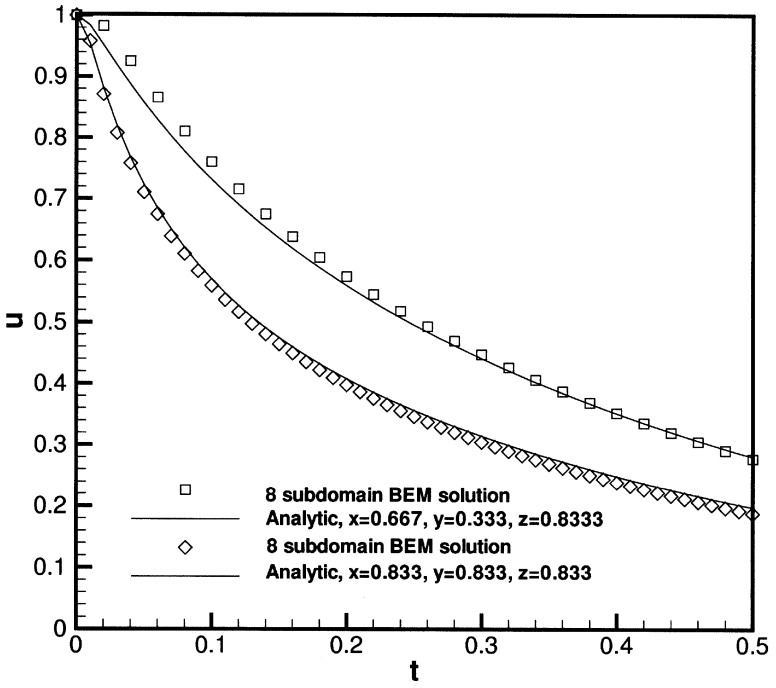

Figure 5. Comparison of BEM subdomain and analytic results at two interior points for benchmark problem 2.

The number of iterations until convergence for the initial time step is shown in Figure 8 for 8, 27, 64, 125, and 216 subdomains with $\gamma = 4.0$ and $\epsilon = 1.0E - 6$. As seen in the figure, the growth rate in the number of iterations with additional subdomains is again sublinear.

The scaled parallel efficiency of the subdomain method can be seen by looking at Table 2, which shows the wall clock time for the various discretizations. As seen in the table, there is very little dependence on the wall clock time per iteration and the number of subdomains. Again, this indicates an almost perfect scaled parallel efficiency. Further, as for the case of the initial time step, the growth in the total number of iterations grows slowly with increasing number of subdomains.

4.3. Problem 3

The third benchmark problem again considers the unit cube comprised of 8 subdomains. However, for this problem the 8 subdomains are comprised of 3 distinct materials, namely, aluminum, tin, and lead. One corner of the cube is placed at the origin of a Cartesian coordinate system and the rest of the cube is in the positive $x - y$, $y - z$, and $z - x$ quadrants. The arrangement of the materials in the cube is shown in Figure 9. Initially, the cube is at temperature 0°, and at time $t = 0$, the back side ($x = 0$) is impulsively raised to 100° while the front side ($x = 1$) is maintained at 0°. The flux on all other sides of the cube is set to $q = 0$ (insulated boundary condition). Because of the material layout, this problem is inherently three-dimensional. Each subdomain is discretized with 294 boundary-element nodes, 54 boundary ele-

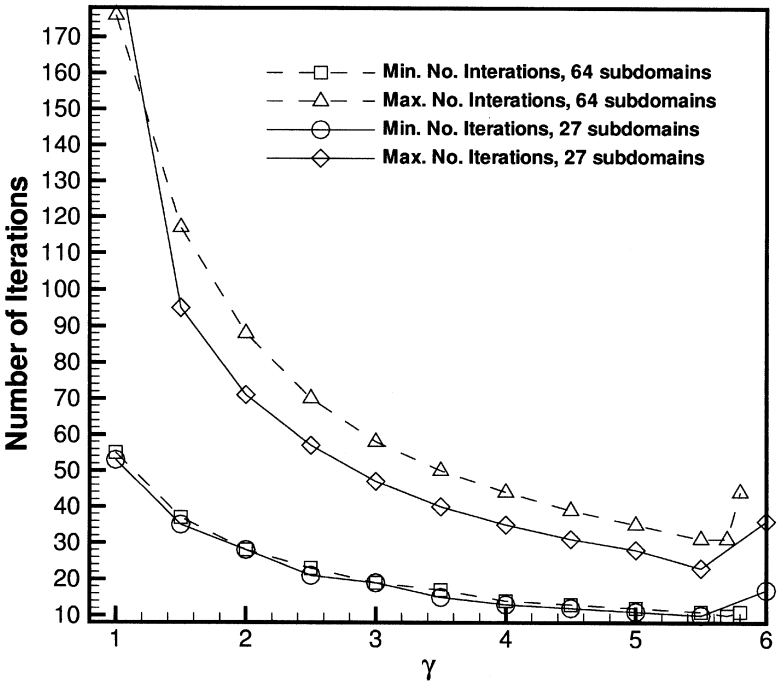

Figure 6. Minimum and maximum number of iterations for the subdomain method to converge as a function of γ for benchmark problem 2 using 27 and 64 subdomains.

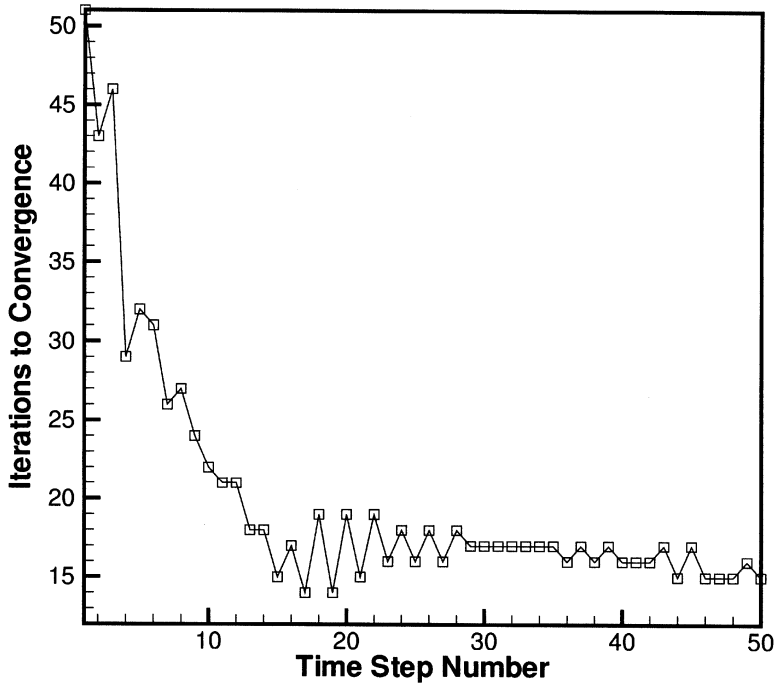

Figure 7. The number of iterations for convergence as a function of the time step for benchmark problem 2 using 64 subdomains and $\gamma = 4.0$.

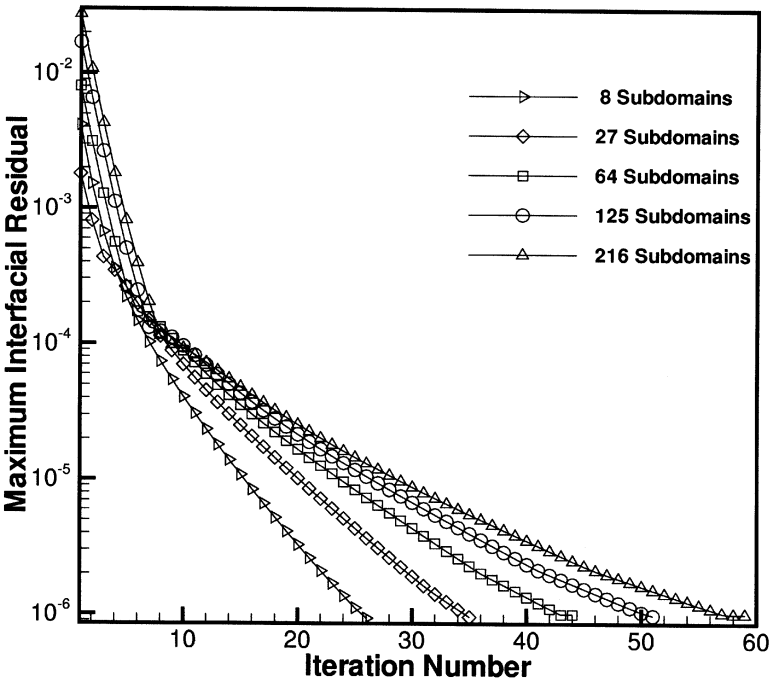

Figure 8. The maximum interfacial residual as a function of the iteration number for the initial time step in benchmark problem 2 for five different discretizations and $\gamma = 4.0$.

ments, and 125 radial basis functions. A time step of 5 s and backward differencing are chosen for this benchmark.

Although the development in Section 3 is for homogeneous domains, only a small change is necessary to extend the development to zoned-homogeneous domains. In particular, Eq. (30) must be modified as follows:

$$q_m^{j,n} = \frac{k^i}{k^j} q_m^{i,n} \tag{32}$$

where k^i and k^j represent the thermal conductivities in the ith and jth subdomains, respectively.

A temperature contour plot at time $t = 1,000$ s for the plane $x = 0.75$ is shown in Figure 10. The zoned-homogeneous regions are fairly easy to discern from the contour plot. The upper right-hand quadrant is at a fairly uniform high temperature,

Table 2. Wall clock times for problem 2 using $\gamma = 4.0$ and $\epsilon = 1.0E - 6$

Number of subdomains	Total number of iterations	Total wall clock time (s)	Wall clock time per iteration (s)
8	640	33.4	0.0522
27	824	55.7	0.0676
64	934	52.0	0.0557
125	1,002	60.0	0.0599

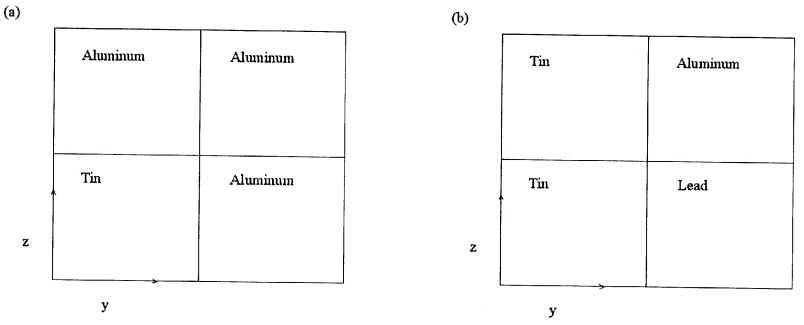

Figure 9. Configuration of materials for benchmark problem 3. (*a*) Four quadrants situated in region $0 < x < 0.5$. (*b*) Four quadrants situated in region $0.5 < x < 1.0$.

since the heat flux is entirely through aluminum in that quadrant. The left-hand side of the figure represents the temperature in the two tin quadrants. The upper left-hand quadrant is warmer than the lower left-hand quadrant because the backplane behind the upper left-hand quadrant is aluminum, whereas the backplane behind the lower left-hand quadrant is tin.

There is no analytic solution for this problem, so a comparison is made between the current boundary-element results and numerical results generated using the commercial finite-element method (FEM) code FIDAP (http://www.fluent.com/software/fidap). The FEM results are generated using 1,728 8-node brick elements, backwards differencing, and a 5 s time step. Hence, each finite-element node

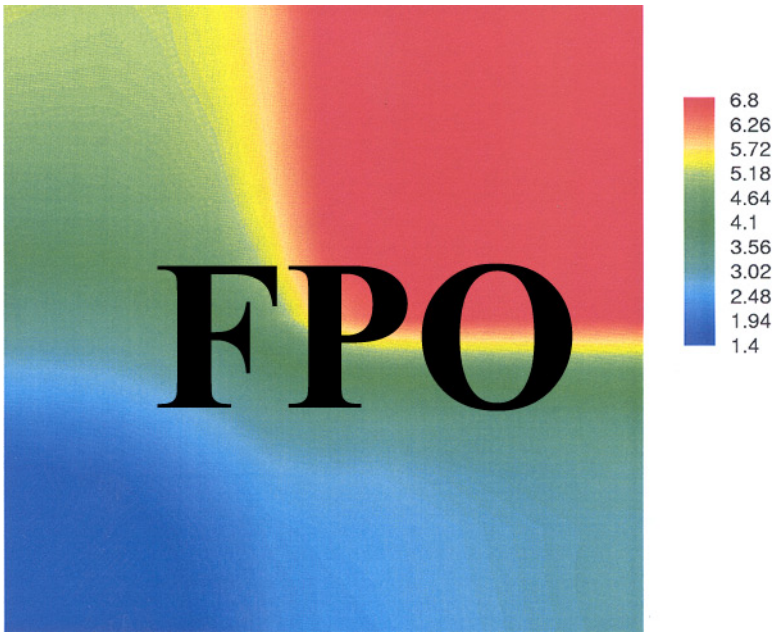

Figure 10. Temperature contour plot at $t = 1,000$ s for the plane defined by $x = 0.75$.

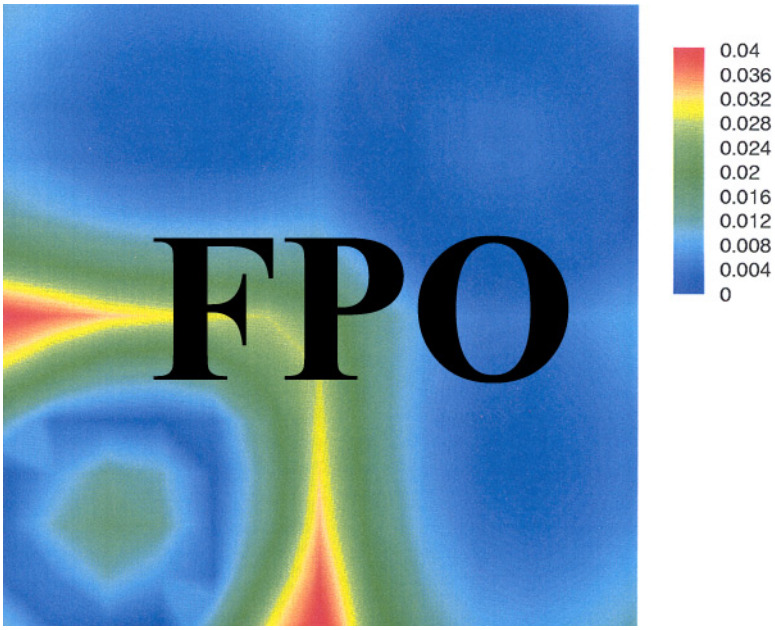

Figure 11. Relative difference contour plot between the BEM and FIDAP results at time $t = 1,000$ s for the plane defined by $x = 0.25$.

corresponds directly to a boundary-element node or radial basis function center. A contour plot of the relative difference between the BEM results and the FIDAP results at time $t = 1{,}000$ s for the plane $x = 0.25$ is shown in Figure 11. The relative difference is defined by

$$\text{Relative difference} = \frac{|\, T_{\text{BEM}} - T_{\text{FIDAP}} \,|}{0.5\,|\,(T_{\text{BEM}} + T_{\text{FIDAP}})|}$$

For the most part, the differences between the FEM and BEM results after 200 time steps are less than 2%. However, differences approaching 4% are seen along the aluminum–tin interfaces. Large temperature gradients exist along the aluminum–tin interface because of the large discontinuity in the conductivities. Nevertheless, the results show reasonable agreement.

5. CONCLUSIONS

A parallel domain decomposition boundary-element method has been developed for three-dimensional diffusion equations. In the current approach, the method of lines is used whereby space is discretized using boundary elements and time is discretized using the generalized trapezoidal (θ-method) rule. At each time step, an approximate particular solution is constructed using a class of multiquadric radial basis functions resulting in a sequence of Poisson problems to trace the time history of the dependent variable.

Parallelism is achieved by breaking up the computational region into subdomains. Each subdomain is mapped to a unique processor. Neumann data are presumed on all subdomain interfaces. The Schwarz Neumann–Neumann iteration is used to achieve continuity of potential and flux along the interfaces. This approach has two key advantages over other parallel methods. First, there is relatively little additional programming required to convert a serial code to a parallel code. Second, relatively little interprocessor communication is required.

Three benchmark problems were considered to assess the accuracy and parallel performance of the method. The convergence of the method was demonstrated for the first two benchmark problems by comparing the BEM results with the analytical solutions. The third benchmark problem extended the capabilities of the BEM formulation to zoned-homogeneous regions. For this benchmark problem, the BEM results were compared with an FEM solution generated using the commercial code FIDAP. For the most part, there was good agreement between the two sets of numerical results. However, larger errors were seen along the tin–aluminum interface. In fact, large temperature gradients exist along the interface because of the discontinuity in both the thermal conductivity and diffusivity. One possible source of error in the BEM solution is caused by the fact that no radial basis function centers are placed on the interface, and hence, the interior approximation may have difficulty in resolving the large thermal gradients near interfaces. This could be remedied in an improved formulation.

Numerical experiment shows that overrelaxation of the Neumann–Neumann iteration scheme, that is, choosing $\gamma > 1.0$ yields the fastest convergence rate. In fact, choosing $2.0 < \gamma < 4.0$ generally yielded good convergence characteristics. However, if γ is chosen too large, the Neumann–Neumann iteration does not converge at all. The mismatch in calculated potential along the interfaces decreases monotonically. Further, the iteration count is seen to go down with time, indicating the efficiency of starting the iteration from the previous time step's converged solution for the interfacial flux.

The scaled parallel efficiency of the domain decomposition method was essentially perfect even though the first two benchmark problems provided an almost worst-case scenario. The reason for this is that many of the subdomains have their entire boundary comprised of interfaces, which results in relatively large communication volume. Nevertheless, the overall communication/computation time is very small, resulting in the excellent scalability.

REFERENCES

1. M. J. Brown and M. S. Ingber, Parallelization of a Vorticity Formulation for the Analysis of Incompressible Viscous Fluid Flows, *Int. J. Numer. Meth. Fluids*, vol. 52, no. 11, pp. 979–999, 2002.
2. D. Curran, M. Cross, and B. A. Lewis, A Preliminary Analysis of Boundary Element Methods Applied to Parabolic Partial Differential Equations, in C. A. Brebbia (Ed.), *New Developments in Boundary Element Methods*, CML Publications, Southampton, UK, 1980.
3. D. Curran, M. Cross, and B. A. Lewis, Solution of Parabolic Differential Equations by the Boundary Element Method Using Discretization in Time, *Appl. Math. Modelling*, vol. 4, pp. 398–400 1980.

4. A. J. Davies and J. Mushtaq, The Domain Decomposition Boundary Element Method on a Network of Transputers, in R. C. Ertekin, C. A. Brebbia, M. Tanaka, and R. Shaw (Eds.), *Proc. Boundary Element Technology XI*, pp. 397–406, Computational Mechanics Publications, Southampton, UK, 1996.
5. W. M. Elleithy and M. Tanaka, Interface Relaxation Algorithms for Coupling the FEM and BEM, in C. A. Brebbia, A. Tadeu, and V. Popov (Eds.), *Proc. Boundary Elements XXIV*, pp. 721–730, WIT Press, Southampton, UK, 2002.
6. C. F. Gerald and P. O. Wheatley, *Applied Numerical Analysis*, Addison-Wesley, Reading, MA, 1994.
7. R. Glowinski, Q. V. Dinh, and J. Periaux, Domain Decomposition Methods for Nonlinear Problems in Fluid Dynamics, *Comput. Meth. Appl. Mech. Eng.*, vol. 40, pp. 27–109, 1983.
8. M. A. Goldberg and C. S. Chen, The Theory of Radial Basis Functions Applied to the BEM for Inhomogeneous Partial Differential Equations, *Bound. Elem. Commun.*, vol. 5, pp. 57–61, 1994.
9. M. A. Goldberg, C. S. Chen, H. Bowman, and H. Power, Some Comments on the Use of Radial Basis Functions in the Dual Reciprocity Method, *Comput. Mech.*, vol. 21, pp. 141–148, 1998.
10. M. S. Ingber and N. Phan-Thien, A Boundary Element Approach for Parabolic Equations Using a Class of Particular Solutions, *Appl. Math. Modelling*, vol. 16, pp. 124–132, 1992.
11. M. S. Ingber, A. A. Mammoli, and M. J. Brown, A Comparison of Domain Integral Evaluation Techniques for Boundary Element Methods, *Int. J. Numer. Meth. Eng.*, vol. 52, no. 4, pp. 417–432, 2001.
12. N. Kamiya, H. Iwase, and E. Kita, Parallel Implementation of Boundary Element Method with Domain Decomposition, *Eng. Anal. Bound. Elem.*, vol. 18, no. 3, pp. 209–216, 1996.
13. C. H. Lai, Diakoptes, Domain Decomposition and Parallel Computing, *Comput. J.*, vol. 37, pp. 840–846, 1994.
14. J. A. Liggett and P. L. F. Liu, Unsteady Flow in Confined Aquifers: A Comparison of Two Boundary Integral Methods, *Water Resources Res.*, vol. 15, pp. 861–866, 1979.
15. N. Mai-Duy, P. Nguyen-Hong, and T. Tran-Cong, A Fast Convergent Iterative Boundary Element Method on a PVM Cluster, *Eng. Anal. Bound. Elem.*, vol. 22, no. 4, pp. 307–316, 1998.
16. G. J. Moridis, Alternative Formulations of the Laplace Transform Boundary Element (LTBE) Numerical Method for the Solution of Diffusion-Type Equations, in C. A. Brebbia and M. S. Ingber (Eds.), *Proc. Boundary Element Technology VII*, pp. 815–833, Computational Mechanics Publications, Southampton, UK, 1992.
17. P. W. Partridge, C. A. Brebbia, and L. C. Wrobel, *The Dual Reciprocity Boundary Element Method*, Elsevier Applied Mechanics, London, UK, 1992.
18. M. J. D. Powell, Radial Basis Functions for Multivariate Interpolation, in J. C. Mason and M. G. Cox (Eds.), *Algorithms for Application*, Clarendon Press, Oxford, UK, 1987.
19. H. Power and R. Mingo, The DRM Subdomain Decomposition Approach to Solve the Two-Dimensional Navier–Stokes System of Equations, *Eng. Anal. Bound. Elem.*, vol. 24, no. 1, pp. 107–119, 2000.
20. H. Power and R. Mingo, The DRM Subdomain Decomposition Approach to Solve the Two-Dimensional Thermal Convection Flow Problems, *Eng. Anal. Bound. Elem.*, vol. 24, no. 1, pp. 121–127, 2000.
21. F. J. Rizzo and D. J. Shippy, A Method of Solution for Certain Problems of Transient Heat Conduction, *AIAA J.*, vol. 8, pp. 2004–2009, 1970.

22. J. N. Shadid, S. A. Hutchinson, H. K. Moffat, G. L. Hennigan, B. Hendrickson, and R. W. Leland, A 65+ Gflop/s Unstructured Finite Element Simulation of Chemically Reacting Flows on the Intel Paragon, *Proc. Supercomputing 94*, IEEE Computer Society Press, Washington DC, 1994.
23. S. R. Subia, M. S. Ingber, and A. K. Mitra, A Comparison of the Semidiscontinuous Element and Multiple Node with Auxiliary Boundary Collocation Approaches for the Boundary Element Method, *Eng. Anal. Bound. Elem.*, vol 15, no. 1, pp. 19–28, 1995.
24. A. E. Taigbenu and J. A. Liggett, Boundary Element Calculations for the Diffusion Equation, *J. Eng. Mech.*, vol. 111, no. 3, pp. 311–328, 1985.
25. L. C. Wrobel and C. A. Brebbia, The Boundary Element Method for Steady-State and Transient Heat Conduction, in R. W. Lewis and and K. Morgan (Eds.), *Numerical Methods in Thermal Problems*, Pineridge Press, Swansea, Wales, 1979.
26. L. C. Wrobel and C. A. Brebbia, Boundary Elements for Non-linear Heat Conduction Problems, *Commun. Appl. Numer. Meth.*, vol. 4, pp. 617–622, 1988.
27. R. Zheng, C. J. Coleman, and N. Phan-Thien, A Boundary Element Approach for Non-homogeneous Potential Problems, *Comput. Math.*, vol. 7, no. 4, pp. 279–288, 1991.