

# A fast multipole accelerated method of fundamental solutions for potential problems

Y.J. Liu<sup>a,\*</sup>, N. Nishimura<sup>b</sup>, Z.H. Yao<sup>c</sup>

<sup>a</sup> Department of Mechanical, Industrial and Nuclear Engineering, University of Cincinnati, Cincinnati, Ohio 45221-0072, USA

<sup>b</sup> Academic Center for Computing and Media Studies, Kyoto University, Kyoto 606-8501, Japan

<sup>c</sup> Department of Engineering Mechanics, Tsinghua University, Beijing 100084, China

Received 15 December 2004; revised 4 March 2005; accepted 14 March 2005

Available online 6 October 2005

---

## Abstract

The fast multipole method (FMM) is a very effective way to accelerate the numerical solutions of the methods based on Green's functions or fundamental solutions. Combined with the FMM, the boundary element method (BEM) can now solve large-scale problems with several million unknowns on a desktop computer. The method of fundamental solutions (MFS), also called superposition or source method and based on the fundamental solutions but without using integrals, has been studied for several decades along with the BEM. The MFS is a boundary meshless method in nature and offers more flexibility in modeling of a problem. It also avoids the singularity of the kernel by placing the source at some auxiliary points off the problem domain. However, like the traditional BEM, the conventional MFS also requires  $O(N^2)$  operations to compute the system of equations and another  $O(N^3)$  operations to solve the system using direct solvers, with  $N$  being the number of unknowns. Combining the FMM and MFS can potentially reduce the operations in formation and solution of the MFS system, as well as the memory requirement, all to  $O(N)$ . This paper is an attempt in this direction. The FMM formulations for the MFS is presented for 2D potential problem. Issues in implementation of the FMM for the MFS are discussed. Numerical examples with up to 200,000 DOF's are solved successfully on a Pentium IV PC using the developed FMM MFS code. These results clearly demonstrate the efficiency, accuracy and potentials of the fast multipole accelerated MFS.

© 2005 Elsevier Ltd. All rights reserved.

**Keywords:** Fast multipole method; Method of fundamental solutions; Potential problems.

---

## 1. Introduction

The method of fundamental solutions (MFS), also called superposition or source method, has been studied for many years along with the boundary integral equation and boundary element method (BIE/BEM, see, e.g. Refs. [1–6]). The MFS uses only the fundamental solution in the construction of the solution of a problem, without using any integrals. It is a natural boundary meshless method and offers several advantages as compared with the BIE/BEM approach. First, meshing a boundary with only points is certainly much easier than with elements. Second, singular integrals are avoided in the MFS (although singularities of the kernel still play a role). Third, programming with the conventional MFS is significantly simplified compared with the BEM. All these advantages with the MFS have attracted continued interests from

researchers. Comprehensive reviews on the MFS for various applications can be found in Refs. [7–9]. Some recent work on the MFS can be found in Refs. [10–13] for potential problems, in Refs. [14,15] for elastostatic problems, and in Ref. [16] for a boundary point method related to the MFS.

However, as in the case of the BEM, the efficiency in solving the MFS equations has been a serious problem for large-scale models. While the finite element method (FEM) has been used routinely to solve models with several millions of degrees of freedom (DOF's), the BEM and MFS has been limited to solving problems with a few thousands DOF's for many years. This is because the conventional BEM and MFS in general produce dense and non-symmetric matrices that, although smaller in sizes, require  $O(N^2)$  operations to compute the coefficients and another  $O(N^3)$  operations to solve the system with direct solvers.

The fast multipole method (FMM) pioneered by Rokhlin and Greengard [17–19] can be used to accelerate the solutions of particle interaction problems and the boundary integral equations. With the FMM, both computing time and memory requirement can be reduced to  $O(N)$ . In the last decade, the fast

\* Corresponding author. Tel.: +513 556 4607; fax: +513 556 3390.

E-mail address: [yijun.liu@uc.edu](mailto:yijun.liu@uc.edu) (Y.J. Liu).

multipole accelerated BEM (FMM BEM) has been developed for solving many large-scale applied mechanics problems. Some of the work on the FMM BEM can be found in Refs. [20–29], which show great promises of the FMM BEM for solving large-scale problems. For example, composite material models containing tens of thousands of fibers [26–28], and models of electromagnetic wave scatterings from a full aircraft at GHz frequencies [29], all have been solved successfully by using the FMM BEM within hours. A comprehensive review of the FMM BEM can be found in Ref. [30]. A boundary node method, which is also a meshless approach and accelerated by the FMM, can be found in Ref. [31].

Because of the similarity of the BEM and MFS, it is natural to apply the fast multipole method to accelerate the solutions of the MFS to develop an FMM MFS. Combining the FMM and MFS can potentially reduce the operations in formation and solution of the MFS system, as well as the memory requirement, all to  $O(N)$ . This will be an important improvement to the MFS for its applications. In this paper, an FMM MFS for 2D potential problems is presented and the developed approach can be extended easily to other cases. The FMM formulations for the MFS are provided and implementation issues of the FMM for the MFS are discussed. Numerical examples with up to 200,000 DOF's are solved successfully on a Pentium IV PC using the developed FMM MFS code. These preliminary results clearly demonstrate the efficiency, accuracy and potentials of the fast multipole accelerated MFS.

This paper is organized as follows: In Section 2, we review the conventional MFS formulation for potential problems. In Section 3, we present the complete formulations and algorithms used in the FMM MFS. In Section 4, we show two example problems solved by using the developed FMM MFS code to demonstrate the efficiencies of the FMM MFS for large-scale problems. We conclude the paper with some discussions in Section 5.

## 2. Formulation of the conventional MFS

Consider the following Laplace equation governing potential problems (for example, a steady-state heat conduction problem) in a 2D domain  $V$  (Fig. 1):

$$\nabla^2 \phi(\mathbf{x}) = 0, \quad \forall \mathbf{x} \in V; \quad (1)$$

under the boundary conditions:

$$\phi(\mathbf{x}) = \bar{\phi}(\mathbf{x}), \quad \forall \mathbf{x} \in S_1; \quad (2)$$

$$q(\mathbf{x}) \equiv \frac{\partial \phi}{\partial n}(\mathbf{x}) = \bar{q}(\mathbf{x}), \quad \forall \mathbf{x} \in S_2; \quad (3)$$

where  $\phi$  is the potential field,  $S = S_1 \cup S_2$  the boundary of  $V$ ,  $n$  the outward normal, and the barred quantities indicate given values on the boundary.

In the method of fundamental solutions, we place  $N$  collocation points ( $\mathbf{x}$ ) on boundary  $S$  and another  $N$  auxiliary or source points ( $\mathbf{y}$ ) outside the domain  $V$  (Fig. 1). We can show

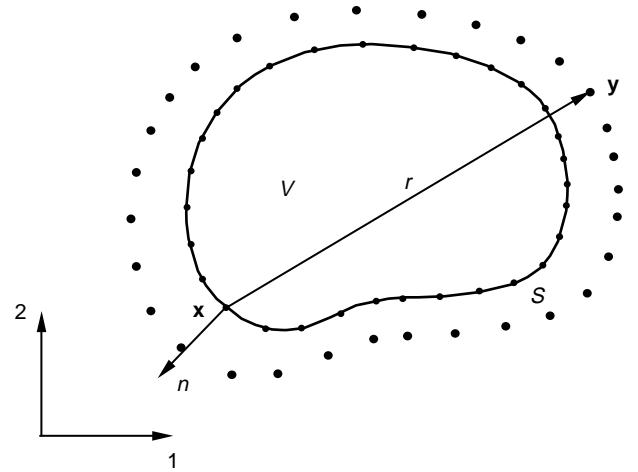


Fig. 1. A domain  $V$  with boundary  $S$ , the collocation point  $\mathbf{x}$  and source point  $\mathbf{y}$ .

that  $\phi$  given by the following expression satisfies Eq. (1):

$$\phi(\mathbf{x}) = \sum_{j=1}^N G(\mathbf{x}, \mathbf{y}_j) \mu_j, \quad (4)$$

where

$$G(\mathbf{x}, \mathbf{y}) = \frac{1}{2\pi} \ln \left( \frac{1}{r} \right) \quad (5)$$

is the fundamental solution for 2D potential problems,  $r$  the distance between  $\mathbf{x}$  and  $\mathbf{y}$  (Fig. 1), and  $\mu_j$  the unknown intensity of the sources at the auxiliary points  $\mathbf{y}_j$ . The boundary conditions in Eqs. (2) and (3) can be satisfied at the collocation points by adjusting  $\mu_j$  at the auxiliary points, that is, imposing

$$\sum_{j=1}^N G(\mathbf{x}_i, \mathbf{y}_j) \mu_j = \bar{\phi}_i, \quad \text{if } \mathbf{x}_i \in S_1; \quad (6)$$

or,

$$\sum_{j=1}^N F(\mathbf{x}_i, \mathbf{y}_j) \mu_j = \bar{q}_i, \quad \text{if } \mathbf{x}_i \in S_2; \quad (7)$$

where

$$F(\mathbf{x}, \mathbf{y}) = \frac{\partial G(\mathbf{x}, \mathbf{y})}{\partial n(\mathbf{x})} = -\frac{1}{2\pi r} \frac{\partial r}{\partial n(\mathbf{x})}. \quad (8)$$

In the conventional MFS (or superposition) approach, the following standard linear system of equations is formed after applying either Eq. (6) or Eq. (7) at all the collocation points  $\mathbf{x}_i$  ( $i = 1, 2, \dots, N$ ):

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{bmatrix} \begin{Bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_N \end{Bmatrix} = \begin{Bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{Bmatrix}, \quad \text{or } \mathbf{A}\boldsymbol{\mu} = \mathbf{b}, \quad (9)$$

where  $\mathbf{A}$  is the coefficient matrix,  $\boldsymbol{\mu}$  the unknown vector and  $\mathbf{b}$  the right-hand side vector. Once all the values of  $\mu_j$  are

determined by solving this equation, the potential at any point inside the domain  $V$  or on the boundary  $S$  can be evaluated using Eq. (4). In the modern least-squares approaches in the MFS, the locations of the sources and the source intensities are solved simultaneously using a nonlinear least-squares approach, which can remove the uncertainty in determining the distance between the sources and the boundary [7]. Also, a smaller number of source points can be used than the number of collocation points, and the system can be solved by using a linear least-squares method. In this paper, however, we focus our attention to the early version of the MFS, that is, Eq. (9), for implementation of the fast multipole method.

The construction of matrix  $\mathbf{A}$  in Eq. (9) requires  $O(N^2)$  operations using the kernel in Eq. (5) or Eq. (8), and the size of the required memory for storing  $\mathbf{A}$  is also  $O(N^2)$ , since  $\mathbf{A}$  is in general a non-symmetric and dense matrix. The solution of the system in Eq. (9) using direct solvers such as Gauss elimination requires  $O(N^3)$  operations. Such characteristics of the conventional MFS have limited its applications in solving large-scale problems.

### 3. The fast multipole method for MFS

The fast multipole method can be employed to accelerate the MFS for solving Eq. (9). The main idea of the FMM is to translate the point-to-point interactions to cell-to-cell interactions by using multipole expansions and translations, where cells can have a hierarchical tree structure. Iterative equation solvers (such as GMRES) are used in the FMM, where matrix–vector multiplications (as for the left-hand side of Eq. (9)) are calculated using fast multipole expansions. Using the FMM, the solution time of a problem can be reduced to order  $O(N)$ . The memory requirement can also be reduced to  $O(N)$  since iterative solvers do not need to store the entire matrix in the memory.

In this section, we present an FMM for the method of fundamental solutions. It turns out that the FMM formulations for the MFS are very similar to those for the BEM, except for the calculations of the moments where the integrals are replaced by summations and the local expansion for the  $F$  kernel sum. For completeness, we provide all the required formulas. Details on the derivations of some of these formulas can be found in Refs. [19,30].

For convenience, let us use the *complex notation*, that is, replace the collocation point  $\mathbf{x}$  and source point  $\mathbf{y}$  by  $z_0 = x_1 + ix_2$  and  $z = y_1 + iy_2$  (with  $i = \sqrt{-1}$  here), respectively (Fig. 2). We write:

$$G(z_0, z) = -\frac{1}{2\pi} \ln(z_0 - z), \quad (10)$$

the real part of which gives the fundamental solution in Eq. (5).

#### 3.1. Multipole expansion (Moments)

Assuming  $z_c$  is a point close to point  $z$  (Fig. 2), that is,  $|z - z_c| \ll |z_0 - z_c|$ , we have:

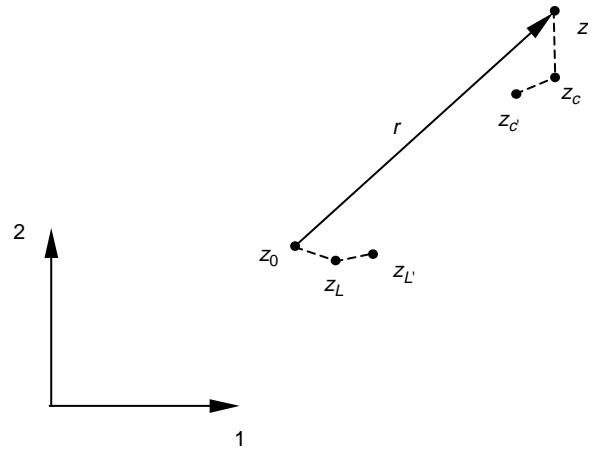


Fig. 2. The related points for the fast multipole expansions.

$$G(z_0, z) = -\frac{1}{2\pi} \ln(z_0 - z) = -\frac{1}{2\pi} \left[ \ln(z_0 - z_c) + \ln\left(1 - \frac{z - z_c}{z_0 - z_c}\right) \right].$$

Applying the Taylor series expansion for the second log term, we obtain:

$$G(z_0, z) = -\frac{1}{2\pi} \left[ \ln(z_0 - z_c) - \sum_{k=1}^{\infty} \frac{1}{k} \frac{(z - z_c)^k}{(z_0 - z_c)^k} \right]. \quad (11)$$

Note that in the  $G$  kernel,  $z_0$  and  $z$  are now separated due to the introduction of the ‘mid point’  $z_c$ , which is a key in the FMM. Considering a group of source point, when the collocation point  $z_0$  is far away from this group, we can evaluate the sum  $\sum_j G(z_0, z_j) \mu_j$  on the left-hand side of Eq. (6) by the following *multipole expansion* using Eq. (11):

$$\sum_j G(z_0, z_j) \mu_j = -\frac{1}{2\pi} \left[ \ln(z_0 - z_c) M_0 + \sum_{k=1}^{\infty} \frac{M_k(z_c)}{(z_0 - z_c)^k} \right], \quad (12)$$

where

$$M_k(z_c) = \begin{cases} \sum_j \mu_j & \text{for } k = 0; \\ -\frac{1}{k} \sum_j (z_j - z_c)^k \mu_j, & \text{for } k \geq 1; \end{cases} \quad (13)$$

are called *moments* about  $z_c$ , which are independent of the collocation point  $z_0$  and only need to be computed once. After these moments are obtained, the terms on the left-hand-side of Eq. (6) can be evaluated readily using Eq. (12) for any point  $z_0$  away from  $z_c$  which will be the center of a *cell* containing all the points  $z_j$ .

Note that the  $F$  kernel given in Eq. (8) is related to the  $G$  kernel by the following expression:

$$F(z_0, z) = n(z_0) \frac{\partial G(z_0, z)}{\partial z_0}, \quad \text{with } n = n_1 + in_2,$$

in the complex notation. Thus, we can derive the following multipole expansion for the sums on the left-hand side of

Eq. (7):

$$\sum_j F(z_0, z_j) \mu_j = -\frac{n(z_0)}{2\pi} \left[ \frac{M_0}{z_0 - z_c} - \sum_{k=1}^{\infty} \frac{k M_k(z_c)}{(z_0 - z_c)^{k+1}} \right], \quad (14)$$

using the same moments given in Eq. (13).

### 3.2. Moment-to-moment translation (M2M)

When the point  $z_c$  is moved to a new location  $z_{c'}$  (Fig. 2), we write:

$$M_k(z_{c'}) = -\frac{1}{k} \sum_j (z_j - z_{c'})^k \mu_j = -\frac{1}{k} \sum_j [(z_j - z_{c'}) + (z_{c'} - z_{c'})]^k \mu_j.$$

Using the binomial formula and rearranging the terms, we obtain:

$$M_k(z_{c'}) = \begin{cases} M_0(z_{c'}), & \text{for } k = 0; \\ -\frac{1}{k} (z_{c'} - z_{c'})^k M_0(z_{c'}) + \sum_{l=1}^k \binom{k-1}{l-1} (z_{c'} - z_{c'})^{k-l} M_l(z_{c'}), & \text{for } k \geq 1. \end{cases} \quad (15)$$

This is the *M2M translation* for the moments when  $z_c$  is moved to  $z_{c'}$ . There is only a finite number of terms in this translation.

### 3.3. Local expansion and moment-to-local translation (M2L)

Suppose  $z_L$  is a point close to the collocation point  $z_0$  (Fig. 2), that is,  $|z_0 - z_L| \gg |z_c - z_L|$ . From the multipole expansion (Eq. (12)), we have:

$$f(z_0) \equiv \sum_j G(z_0, z_j) \mu_j = -\frac{1}{2\pi} \left[ \ln(z_0 - z_c) M_0 + \sum_{k=1}^{\infty} \frac{M_k(z_c)}{(z_0 - z_c)^k} \right]. \quad (16)$$

Expanding  $f(z_0)$  about point  $z_L$  using the Taylor series expansion, we can derive the following *local expansion*:

$$\sum_j G(z_0, z_j) \mu_j = \sum_{l=0}^{\infty} L_l(z_L) (z_0 - z_L)^l, \quad (17)$$

where the coefficients are given by the following *M2L translation*:

$$L_l(z_L) = \begin{cases} -\frac{1}{2\pi} \left[ \ln(z_L - z_c) M_0 + \sum_{k=1}^{\infty} \frac{M_k(z_c)}{(z_L - z_c)^k} \right], & \text{for } l = 0; \\ -\frac{1}{2\pi} \left[ \frac{(-1)^{l+1}}{l} \frac{M_0}{(z_L - z_c)^l} + \frac{(-1)^l}{(z_L - z_c)^l} \sum_{k=1}^{\infty} \binom{l+k-1}{k-1} \frac{M_k(z_c)}{(z_L - z_c)^k} \right], & \text{for } l \geq 1. \end{cases} \quad (18)$$

The local expansion for the  $F$  kernel sum is given by:

$$\sum_j F(z_0, z_j) \mu_j = n(z_0) \sum_{l=1}^{\infty} L_l(z_L) (z_0 - z_L)^{l-1}. \quad (19)$$

### 3.4. Local-to-local translation (L2L)

If the point for local expansion is moved from  $z_L$  to  $z_{L'}$  (Fig. 2), we have the following expression using an  $n$ -term local expansion from Eq. (17):

$$\begin{aligned} \sum_j G(z_0, z_j) \mu_j &= \sum_{l=0}^n L_l(z_L) (z_0 - z_L)^l \\ &= \sum_{l=0}^n L_l(z_L) [(z_0 - z_{L'}) + (z_{L'} - z_L)]^l. \end{aligned}$$

Applying the binomial formula and the relation  $\sum_{l=0}^n \sum_{m=0}^l = \sum_{m=0}^n \sum_{l=m}^n$ , we obtain:

$$\sum_j G(z_0, z_j) \mu_j = \sum_{l=0}^n L_l(z_{L'}) (z_0 - z_{L'})^l, \quad (20)$$

where the new coefficients are given by the following *L2L translation*:

$$L_l(z_{L'}) = \sum_{m=l}^n \binom{m}{l} (z_{L'} - z_L)^{m-l} L_m(z_L), \quad \text{for } l \geq 0. \quad (21)$$

Note that the M2M and L2L translations for the  $F$  kernel sum remain the same.

### 3.5. FMM algorithms for the MFS

The algorithms or procedures for the FMM MFS are very similar to those for the FMM BEM (see, e.g. Ref. [30]). However, there is a major difference, which stems from the fact that the collocation points and source points are two different sets of points in the MFS. Therefore, two separate tree structures are required for the translations. The major steps in the FMM MFS can be described as follows:

**Step 1. Discretization.** For a given problem, discretize the boundary  $S$  as usual as in the conventional MFS, that is, place a certain number of collocation points on  $S$  and source points outside the domain, as shown in Fig. 1.

**Step 2. Determine the hierarchical cell structures.** Consider a square that covers all the collocation and source points. Call this square the cell of level 0 (see Fig. 3). To form the cell structure for the *source* points, we start dividing this *parent* cell into four equal *child* cells of level 1. Continue dividing in this way, that is, take a parent cell of level  $l$  and divide it into four child cells of level  $L+1$ . Stop dividing a cell if the number of points in that cell is less than a specified number (this number is 1 in the example shown in Fig. 3).

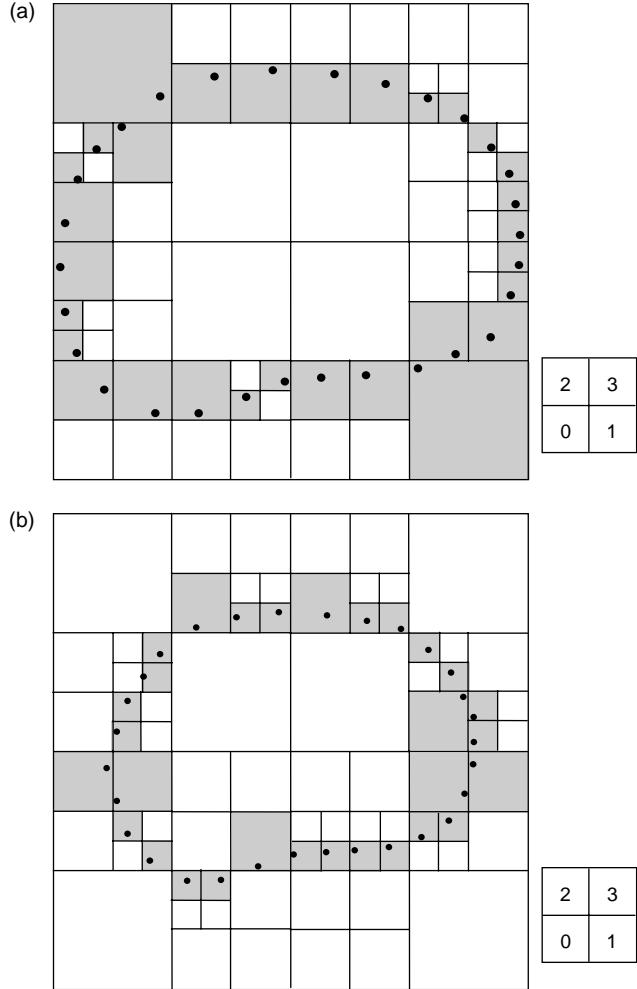


Fig. 3. Construction of the cell structures for (a) source points and (b) collocation points.

A cell having no child cells is called a *leaf* (the shaded cells in Fig. 3). A hierarchical cell or *quad-tree* structure covering all the source points can be formed using this procedure (Fig. 3(a)). A similar cell structure for the *collocation* points can be formed through the same procedure (Fig. 3(b)).

**Step 3. Upward pass.** Consider the quad tree for the source points and compute the moments on all cells, at all levels with  $l \geq 2$ , and for up to  $p$  terms, tracing the tree structure upward. For a leaf, Eq. (13) is applied directly for all the points in the leaf (with  $z_c$  being the centroid of the leaf). For a parent cell, the moment is calculated by summing the moments on its four child cells using the M2M translation, that is, Eq. (15), in which  $z_{c'}$  is the centroid of the parent cell and  $z_c$  the centroid of a child cell (Fig. 2).

**Step 4. Downward pass.** Two cells are said to be *adjacent cells at level  $l$*  if they have at least one common vertex. Two cells are said to be *well separated at level  $l$*  if they are not adjacent at level  $l$  but their parent cells are adjacent at level  $l-1$ . The list of all the well-separated cells from a level  $l$  cell  $C$  is called

the *interaction list* of  $C$ . Cells are called to be *far cells* of  $C$  if their parent cells are not adjacent to the parent cell of  $C$ . We now consider the quad tree for the collocation points and compute the local expansion coefficients on all cells starting from level 2 and tracing the tree structure downward to all the leaves. The local expansion coefficient associated with a cell  $C$  is the sum of the contributions from the cells in the interaction list of cell  $C$  and from all the far cells. The former is calculated by using the M2L translation, Eq. (18), with moments associated with cells in the interaction list, and the latter is calculated by using the L2L translation, Eq. (21), for the parent cell of  $C$  with the expansion point being shifted from the centroid of  $C$ 's parent cell to that of  $C$ . For a cell  $C$  at level 2, we use only M2L translation to compute the coefficients of the local expansion.

**Step 5. Evaluation of the sums in Eq. (6) or (7).** Suppose the values of  $\mu_j$  are given and the collocation point  $z_0$  is in leaf  $C$ . We compute the contributions from source points in leaf  $C$  and its adjacent cells directly as in the conventional MFS. Contributions from all other cells (cells in the interaction list of  $C$  and far cells) are computed by using the local expansion, that is, Eq. (17) or Eq. (19). This is done by using the local expansion coefficients for cell  $C$ , which have been computed in Step 4, shifting the expansion point from the centroid of  $C$  back to the collocation point  $z_0$ , and evaluating the local expansion.

**Step 6. Iterations of the solution.** We update the unknown vector in the system  $\mathbf{A}\boldsymbol{\mu} = \mathbf{b}$  corresponding to Eq. (6) or Eq. (7), and continue at Step 3 for the matrix and unknown vector multiplication until the solution converges within a given tolerance using the GMRES solver.

When the problem size ( $N$ ) is large, the estimated cost of the whole process is  $O(N)$  with  $N$  being the number of the collocation points, if the number of terms  $p$  in the multipole expansions and the number of points in a leaf are kept constant (see Ref. [30]).

### 3.6. Preconditioners for the FMM MFS

Selection of a good pre-conditioner for the FMM BEM is crucial for its convergence and computing efficiency. It is even more so with the FMM MFS, since the conditioning of the MFS system is in general less suitable for using iterative solvers due to the loss of the diagonal dominance. In this study, we use a block diagonal preconditioner, which is formed on each leaf using direct evaluation of the fundamental solution with the collocation points within that leaf and their corresponding source points, which may reside outside the leaf. This preconditioner may not be efficient if the number of collocation points in a leaf is large (for example, more than 500). Using other forms of the pre-conditioners [32] is possible in the FMM

MFS and will be an important subject in improving the efficiency of the FMM MFS in other applications.

#### 4. Test problems

We present two numerical examples to demonstrate the feasibility, accuracy and efficiency of the FMM MFS for 2D potential problems. We compare the efficiency and accuracy of the developed FMM MFS with those of the conventional MFS. All the computations were done on a Pentium IV PC with a 2.4 GHz CPU and 1 GB RAM. The largest model with 200,000 equations run for less than 550 s on this PC.

##### 4.1. An annular region

A simple potential (heat conduction) problem in an annular region (Fig. 4) is considered, for which the analytical solution is available. The used parameters are:  $a = 1$ ,  $b = 1$ ,  $\phi_a = 100$ , and  $q_b = 200$ . This gives  $\phi_b = 377.258872$  and  $q_a = -400.0$  in the analytical solution on the outer and inner surfaces, respectively. The source points are placed outside the domain, along the normal directions of the boundary at their corresponding collocation points, and with a distance of around 0.01 away from the boundary.

We discretize the inner and outer boundaries with the same number of points and run both the FMM MFS code and a conventional MFS code. The conventional MFS code uses a direct solver (LAPACK) for solving the linear system. For the FMM MFS, the numbers of terms for both moments and local expansions were set to 15 and the tolerance for convergence of the solution to  $10^{-6}$ . The FMM MFS results converged within 10 iterations for this tolerance.

The results of  $\phi_b$  and  $q_a$  for this problem using both the FMM MFS and the conventional MFS as the total number of points increase from 36 to 9600 are shown in Table 1. The results for both the FMM MFS and conventional MFS converge quickly to the exact solution for the case with 360 collocation points. For the larger models (e.g. with 9600 points), the results deviate slightly

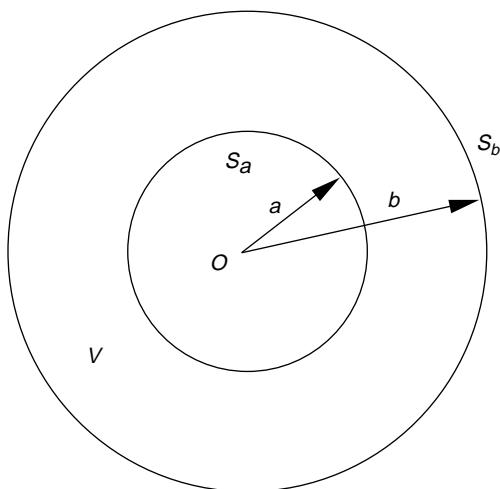


Fig. 4. An annular region  $V$ .

Table 1  
Computed potential and normal derivative for the annular region

N	$q_a$		$\phi_b$	
	Conventional MFS	FMM MFS	Conventional MFS	FMM MFS
36	-417.93852	-417.93877	401.85996	401.85994
72	-408.02599	-408.02564	385.40503	385.40503
360	-399.95961	-401.55737	377.29202	378.43467
720	-399.98451	-400.64512	377.26355	377.72789
1440	-399.99572	-400.65444	377.26004	377.71908
2400	-399.88298	-400.66282	377.25852	377.71617
4800	-399.98733	-400.67203	377.25928	377.71425
7200	-399.99616	-400.63663	377.25896	377.71365
9600	-399.89316	-400.63702	377.25741	377.77159
Exact	-400.0		377.25887	

from the exact solution for both FMM MFS and conventional MFS, which may be caused by truncation errors. In general, the FMM MFS is found to be equally accurate as the conventional MFS with moderate values for the parameters in the FMM MFS. The CPU times used for both approaches in these calculations are plotted in Fig. 5. For comparison, the CPU time using a conventional BEM code with constant elements is also plotted in this figure. The figure shows significant advantage of the FMM MFS in the savings compared with the conventional MFS as well as the conventional BEM. For example, for the largest model with 9600 points, the FMM MFS used only about 62 CPU seconds, while both the conventional MFS and BEM used more than 1400 s.

It was noticed that the convergence of the FMM MFS solutions is sensitive to the distance between the source points and the boundary. For example, for the model with 360 points for the annular region, converged results can be obtained with about 10 iterations if the distance between the source points and boundary is between 0.01 and 0.1. Smaller distances (less than 0.01) would continue to give converged results but the accuracies of the solutions start to deteriorate, which may be caused by the singularity of the kernel. Larger distances (greater than 0.1) would fail to give converged results. This convergence issue needs further investigation for the FMM MFS, as will be discussed in Section 5.

##### 4.2. Thermal analysis of a square plate with circular holes

Thermal conductivity of a square plate with uniformly distributed circular holes, or a perforated plate (Fig. 6), is studied next using the FMM MFS. Several models of the plate, with increasing dimensions and thus the number of holes, are considered. Each model of the plate has a dimension of  $m \times m$ , containing a total of  $2m \times 2m$  holes, with  $m = 1, 2, 3, \dots, 15$ , and 20. The radii of the holes are the same (0.1) for each model that gives a total ‘volume’ fractions of the holes equal to 12.47%. Each hole is discretized with 120 points and the outer boundary of the plate is discretized with  $400m$  points. Thus, the largest model (Fig. 6) with 1600 holes ( $m=20$ ) has a total DOF’s of 200,000. For the boundary conditions,  $\phi=0$  and  $m$  are imposed on the left and right edges, respectively, while

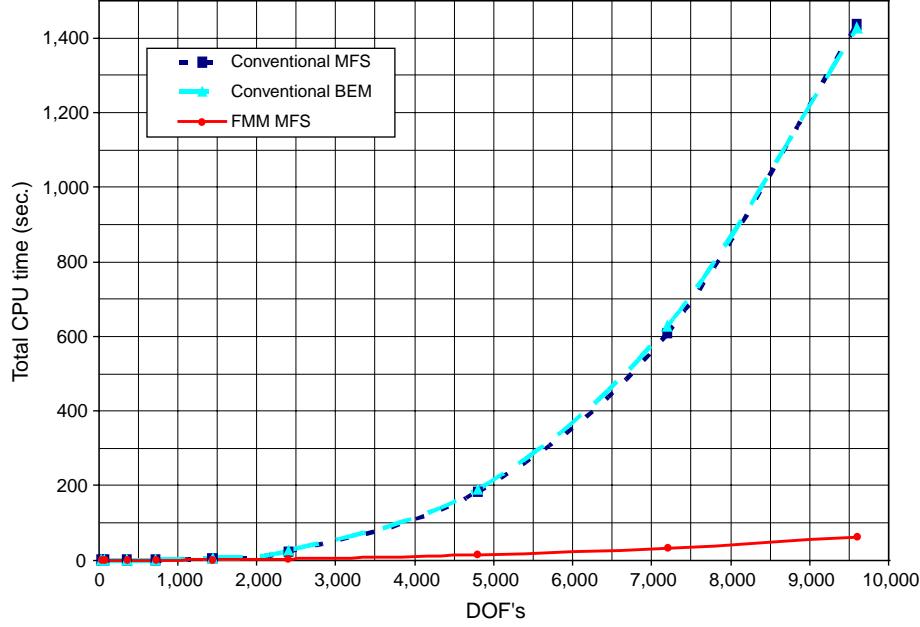


Fig. 5. Comparison of the CPU time for the annular-region problem.

$q=0$  is imposed on the top and bottom edges, and on the edges of all the holes. Using this model, the effective thermal conductivity of the perforated plate can be calculated by the formula [26]:

$$k_{x(\text{eff})} = -\frac{q_{x(\text{ave})}L}{\Delta\phi},$$

where  $k_{x(\text{eff})}$  is the effective thermal conductivity of the plate,  $q_{x(\text{ave})}$  the averaged value of the heat flux,  $\Delta\phi$  the increase of the temperature, and  $L$  the dimension of the square model, all in the  $x$ -direction.

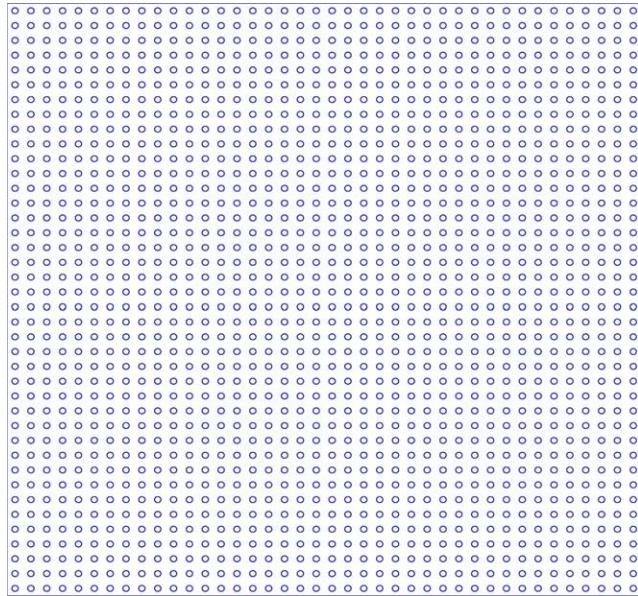


Fig. 6. A model of the square plate with 1600 uniformly distributed circular holes.

**Table 2** shows the calculated effective thermal conductivity  $k_{x(\text{eff})}$  ( $xk_0$ , with  $k_0$  being the reference thermal conductivity of the plate without holes) using both the conventional MFS (which can only solve models up to about 10,000 DOF's) and the FMM MFS. Since the holes are uniformly distributed, theoretically speaking, the calculated  $k_{x(\text{eff})}$  should be independent of the sizes of the models used. The results indeed support this assertion, as shown in **Table 2**. In this example, the numbers of terms for both expansions were set to 15 and the tolerance for convergence of the solution to  $10^{-6}$  in the FMM MFS. **Fig. 7** shows the CPU time for the FMM MFS in solving these relatively large models, as compared with the conventional MFS. It is seen that the total CPU time increases almost linearly with the increase of the total DOF's for the FMM MFS, which clearly demonstrates the  $O(N)$  efficiency of the FMM MFS in solving large-scale problems. The largest model with 200,000 DOF's took less than 550 s on the Pentium IV PC. Random distributions of the holes were also attempted. However, it was found that the convergence of the solutions were rather slow. More general cases of perforated plates with

**Table 2**  
The effective thermal conductivity ( $xk_0$ ) of the perforated plate

Model (Number of holes included)	DOF's	Conventional MFS	FMM MFS
2×2	880	0.77298465	0.77298467
4×4	2720	0.77698081	0.77698068
6×6	5520	0.77595073	0.77595058
8×8	9280	0.77583410	0.77583403
12×12	19,680	—	0.77496649
20×20	52,000	—	0.77411789
30×30	114,000	—	0.77545882
40×40	200,000	—	0.77540569

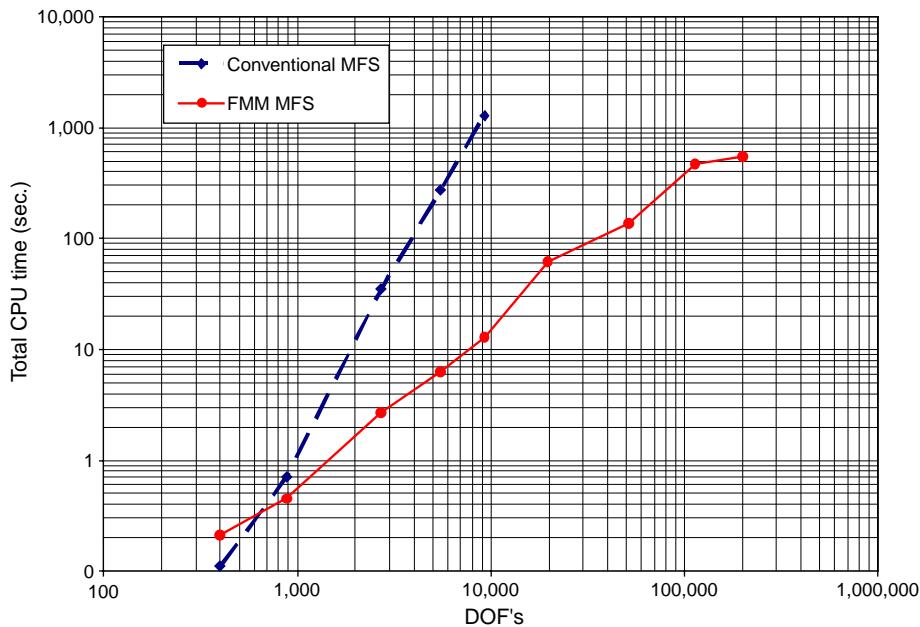


Fig. 7. CPU time used for solving the plate-hole models.

randomly distributed holes can be investigated in the future with a more effective pre-conditioner.

## 5. Discussions

A fast multipole accelerated method of fundamental solutions (FMM MFS) is presented in this paper for 2D potential problems. Complete formulations and implementation details for the FMM MFS are provided. Numerical examples are presented which clearly demonstrate the efficiencies of the FMM MFS for solving large-scale problems. The approach presented in this paper can be extended readily to 2D vector (elastostatic) problems, 3D scalar and vector problems, as well as time-harmonic and dynamic problems.

More research need to be done to improve the developed FMM MFS, regarding, for example, convergence of the solutions, optimization of the tree structures, and calculations of the domain and boundary values of the potential and its derivatives.

The selection of the distance between the boundary and the source points has been a nuisance associated with the conventional MFS ever since its beginning, but it seems even more so with the FMM MFS. This may be caused by the use of iterative solver in the FMM and the pre-conditioner. It was found that smaller values of the distances are needed for the convergence of the FMM MFS solutions. This may due to the fact that the diagonal dominance in the matrix  $\mathbf{A}$  may be more pronounced when the source points are closer to the boundary and thus the pre-conditioner is more effective. In the cases of larger values of the distances, the convergence of the FMM MFS solutions is very slow or fails to converge at all. This may be caused by the loss of the diagonal dominance of the matrix and thus the pre-conditioner is less effective. The convergence issue for the FMM MFS requires further investigations

and better pre-conditioners need to be devised. A better solution may be to employ the modern least-squares approaches in the FMM MFS, where the locations of the source points are determined along with the solution of the problem in a least-squares approach [10–13].

The algorithms of the FMM MFS can also be improved. For example, two quad-tree structures were used in this work, one for the collocation points (in the downward pass) and another for the source points (in the upward pass). It is will be advantageous to use just one tree structure for both upward and downward passes for the MFS. This can save the memory requirement and make the code more efficient. The calculation of the boundary and domain values of the potential and its derivatives were done using the direct evaluation method in this work, that is, using Eq. (4) and its derivatives directly. It is also possible to apply the FMM to speed up such evaluations, which can be computing intensive when the number of data points is comparable to that of the sources. However, this may require the construction of yet another quad-tree structure for such data points. All these issues deserve further investigations based on the promises indicated by this study on the FMM MFS.

## Acknowledgements

The first author (Y.J.L.) would like to thank the JSPS fellowship of Japan, the Chunhui Fellowship of China, and the Department of Engineering Mechanics of the Tsinghua University.

## References

- [1] Jaswon MA. Integral equation methods in potential theory. I. Proc Royal Soc London A 1963;275:23–32.
- [2] Rizzo FJ. An integral equation approach to boundary value problems of classical elastostatics. Q Appl Math 1967;25:83–95.

- [3] Mukherjee S. Boundary element methods in creep and fracture. New York: Applied Science Publishers; 1982.
- [4] Cruse TA. Boundary element analysis in computational fracture mechanics. Dordrecht, The Netherlands: Kluwer Academic Publishers; 1988.
- [5] Brebbia CA, Dominguez J. Boundary elements—an introductory course. New York: McGraw-Hill; 1989.
- [6] Banerjee PK. The boundary element methods in engineering. 2nd ed. New York: McGraw-Hill; 1994.
- [7] Fairweather G, Karageorghis A. The method of fundamental solutions for elliptic boundary value problems. *Adv Comput Math* 1998;9(1/2):69–95.
- [8] Golberg MA, Chen CS. The method of fundamental solutions for potential, Helmholtz and diffusion problems. In: Golberg MA, editor. *Boundary Integral Methods: Numerical and Mathematical Aspects*. Boston: Computational Mechanics Publications; 1998. p. 103–76.
- [9] Fairweather G, Karageorghis A, Martin PA. The method of fundamental solutions for scattering and radiation problems. *Eng Anal Boundary Elements* 2003;27(7):759–69.
- [10] Berger JR, Karageorghis A. The method of fundamental solutions for heat conduction in layered materials. *Int J Numer Meth Eng* 1999;45(11): 1681–94.
- [11] Ramachandran PA. Method of fundamental solutions: singular value decomposition analysis. *Commun Numer Meth Eng* 2002;18(11): 789–801.
- [12] Smyrlis Y-S, Karageorghis A. A matrix decomposition MFS algorithm for axisymmetric potential problems. *Eng Anal Boundary Elements* 2004; 28(5):463–74.
- [13] Mitic P, Rashed YF. Convergence and stability of the method of meshless fundamental solutions using an array of randomly distributed sources. *Eng Anal Boundary Elements* 2004;28(2):143–53.
- [14] Poullikkas A, Karageorghis A, Georgiou G. The method of fundamental solutions for three-dimensional elastostatics problems. *Comput Struct* 2002;80(3–4):365–70.
- [15] de Medeiros GC, Partridge PW, Brandao JO. The method of fundamental solutions with dual reciprocity for some problems in elasticity. *Eng Anal Boundary Elements* 2004;28(5):453–61.
- [16] Dong CY, Lee KY. Stress analysis of an infinite anisotropic elastic medium containing inclusions using the boundary point method. *Eng Anal Boundary Elements* 2004;28(11):1293–302.
- [17] Rokhlin V. Rapid solution of integral equations of classical potential theory. *J Comput Phys* 1985;60:187–207.
- [18] Greengard LF, Rokhlin V. A fast algorithm for particle simulations. *J Comput Phys* 1987;73(2):325–48.
- [19] Greengard LF. The rapid evaluation of potential fields in particle systems. Cambridge: MIT Press; 1988.
- [20] Peirce AP, Napier JAL. A spectral multipole method for efficient solution of large-scale boundary element models in elastostatics. *Int J Numer Meth Eng* 1995;38:4009–34.
- [21] Gomez JE, Power H. A multipole direct and indirect BEM for 2D cavity flow at low Reynolds number. *Eng Anal Boundary Elements* 1997;19: 17–31.
- [22] Fu Y, Klimkowski KJ, Rodin GJ, Berger E, Browne JC, Singer JK, Geijn RAVD, Vemaganti KS. A fast solution method for three-dimensional many-particle problems of linear elasticity. *Int J Numer Meth Eng* 1998;42:1215–29.
- [23] Nishimura N, Yoshida K, Kobayashi S. A fast multipole boundary integral equation method for crack problems in 3D. *Eng Anal Boundary Elements* 1999;23:97–105.
- [24] Mammoli AA, Ingber MS. Stokes flow around cylinders in a bounded two-dimensional domain using multipole-accelerated boundary element methods. *Int J Numer Meth Eng* 1999;44:897–917.
- [25] Yao Z, Kong F, Wang H, Wang P. 2D Simulation of composite materials using BEM. *Eng Anal Boundary Elements* 2004;28(8):927–35.
- [26] Nishimura N, Liu YJ. Thermal analysis of carbon-nanotube composites using a rigid-line inclusion model by the boundary integral equation method. *Comput Mech* 2004;35(1):1–10.
- [27] Liu YJ, Nishimura N, Otani Y, Takahashi T, Chen XL, Munakata H. A fast boundary element method for the analysis of fiber-reinforced composites based on a rigid-inclusion model. *J Appl Mech* 2005;72(1): 115–28.
- [28] Liu YJ, Nishimura N, Otani Y. Large-scale modeling of carbon-nanotube composites by the boundary element method based on a rigid-inclusion model. *Comput Mater Sci* 2005;34(2), 173–187.
- [29] Chew WC, Chao HY, Cui TJ, Lu CC, Ohnuki S, Pan YC, Song JM, Velampampil S, Zhao JS. Fast integral equation solvers in computational electromagnetics of complex structures. *Eng Anal Boundary Elements* 2003;27(8):803–23.
- [30] Nishimura N. Fast multipole accelerated boundary integral equation methods. *Appl Mech Rev* 2002;55(4):299–324.
- [31] Zhang J, Tanaka M, Endo M. The hybrid boundary node method accelerated by fast multipole method for 3D potential problems. *Int J Numer Meth Eng* [in press].
- [32] Wang HT, Yao ZH, Wang PB. Application of preconditioners for fast multipole boundary element methods for multi-domain elastostatics. *Eng Anal Boundary Elements* [in review].