A MULTIPOLE BOUNDARY ELEMENT METHOD FOR TWO DIMENSIONAL ELASTOSTATICS *

YOSIHIRO YAMADA† ‡ AND KEN HAYAMI†

Abstract. The boundary element method requires discretization only on the boundary. However, $O(N^3)$ work and $O(N^2)$ memory is usually required, where N is the number of boundary elements. To avoid these problems, Rokhlin proposed the multipole method for the potential problem. This paper extends the method to the two dimensional elastostatic problem. Formulations are given for the Dirichlet, Neumann and mixed boundary value problems, including the evaluation of the internal stress. The method requires $O(N \log N)$ work and memory. Theoretical error estimates for the multipole expansions are also derived. Numerical examples demonstrate the efficiency of the proposed method compared to the standard techniques.

Key Words. boundary element method, two dimensional elastostatics, multipole method, clustering method, iterative method, GCR, Bi-CGSTAB

AMS subject classification. 65N38, 65F10

1. Introduction. The Boundary Element Method (BEM) [1] has become a popular simulation method in science and engineering in recent years. Its main advantages are the following:

1. Discretization is required only on the boundary of the region being considered. This also holds when the region is infinite, making the method even more attractive.
2. The field variable and its derivatives are calculated accurately using all the data on the boundary, instead of local interpolation or numerical differentiation.
3. Discontinuities such as cracks may be treated accurately by the use of singular fundamental solutions.

However, the method is not necessarily computationally efficient for large scale problems. This is because the method involves solving a system of linear equations for the unknowns on the boundary, where the matrix is dense and nonsymmetric. If the boundary is discretized into N boundary elements, the standard LU decomposition procedure would require $O(N^3)$ computation and $O(N^2)$ memory, which becomes prohibitive as N becomes large, as in three dimensional problems, and even for two dimensional problems if the geometry is complex, as in regions with many holes.

The last decade has seen the development of efficient iterative solvers for nonsymmetric systems of linear equations such as GMRES [5] and Bi-CGSTAB [6] in conjunction with the use of efficient preconditioners. Although these methods have been mainly addressed for sparse systems arising in the finite difference or finite element methods, they may also be applied to BEM. This would reduce the amount of computation, provided the number of iterations is less than $O(N)$. However, the memory required to compute the matrix vector product is still $O(N^2)$, and it is this fact, more than the amount of computation, that hinders the use of BEM for large scale problems, since the actual elapsed time is governed by the memory size.

* This work was supported by the Grant-in-Aid for Scientific Research (C) from the Ministry of Education, Science and Culture, Japan.

† Department of Mathematical Engineering and Information Physics, Faculty of Engineering, University of Tokyo, 7-3-1, Hongo, Bunkyo-ku, Tokyo 113, Japan (hayami@simplex.t.u-tokyo.ac.jp).

‡ Presently at Shimadzu Corporation, Kyoto, Japan.

In order to resolve this problem, Rokhlin proposed the multipole expansion method for the two dimensional potential problem [4]. The method computes the above matrix vector product approximately (to any required accuracy), with $\mathrm{O}(N \log N)$ computation and memory, by clustering the effect of the distant boundary elements using multipole expansions. Greengard et al. improved the method to $\mathrm{O}(N)$ (the fast multipole method), and generalized it to three dimensional potential problems [3] and multiply connected domains [2].

In this paper, we propose a multipole expansion method for the two dimensional elastostatic problem. The elastostatic problem is computationally more intensive compared to the potential problem, since it requires d-unknowns per boundary element for the d-dimensional problem, instead of one. Moreover, the fundamental solutution of the elastostatic problem is more complicated compared to the potential problem, so that a new formulation is required in order to apply the multipole method.

2. Boundary element formulation of two dimensional elastostatics. [1] Throughout this paper, the Greek characters α, β and γ will be used to indicate the coordinate components, and the Roman characters i and j will be used to represent the boundary elements, i.e. $\alpha, \beta, \gamma = 1, 2$, and $i, j = 1, 2, \ldots, N$, where N is the total number of boundary elements.

Let $\boldsymbol{x}$ be a point in a domain Ω in $\mathbf{R}^2$, i.e. $\boldsymbol{x} \equiv (x_1, x_2)^{\mathrm{T}} \in \Omega$. The partial differential equation governing the two dimensional elastostatic problem is given by

$$\sum_{\beta} \frac{\partial \sigma_{\alpha\beta}(\boldsymbol{x})}{\partial x_\beta} = b_\alpha(\boldsymbol{x}), \qquad \alpha, \beta = 1, 2, \tag{1}$$

where $(\sigma_{\alpha\beta})$ represents the stress tensor. To simplify the argument, the body force term b_α is ignored in the following. The relation between the stress tensor and the displacement

$$\boldsymbol{u}(\boldsymbol{x}) \equiv \begin{pmatrix} u_1(\boldsymbol{x}) \\ u_2(\boldsymbol{x}) \end{pmatrix}$$

is given by

$$\sigma_{\alpha\beta} = \lambda \delta_{\alpha\beta} \sum_{\gamma} \epsilon_{\gamma\gamma} + 2\mu \epsilon_{\alpha\beta}, \qquad \epsilon_{\alpha\beta} \equiv \frac{1}{2} \left(\frac{\partial u_\alpha}{\partial x_\beta} + \frac{\partial u_\beta}{\partial x_\alpha} \right), \tag{2}$$

where $\epsilon_{\alpha\beta}$ is the strain tensor, $\delta_{\alpha\beta}$ is the Kronecker's delta, and λ, μ are Lamé's constants.

The boundary condition for (1) is given by

$$u_\alpha = \bar{u}_\alpha \qquad \boldsymbol{x} \in \Gamma_\alpha^{(\mathrm{D})}, \tag{3}$$

$$p_\alpha \equiv \sum_{\beta} \sigma_{\alpha\beta} n_\beta = \bar{p}_\alpha \qquad \boldsymbol{x} \in \Gamma_\alpha^{(\mathrm{N})}, \tag{4}$$

where

$$\Gamma_\alpha^{(\mathrm{D})} + \Gamma_\alpha^{(\mathrm{N})} = \Gamma \equiv \partial\Omega,$$

$\boldsymbol{n}$ is the unit outward normal vector at $\boldsymbol{x} \in \Gamma$, and $\boldsymbol{p} \equiv p_\alpha$ is the traction at $\boldsymbol{x} \in \Gamma$.

Let $u^*_{\alpha\beta}(\boldsymbol{x};\boldsymbol{x}_0)$ be the fundamental solution of (1) in terms of the displacement, and let $p^*_{\alpha\beta}(\boldsymbol{x};\boldsymbol{x}_0)$ be the traction caused by $u^*_{\alpha\beta}(\boldsymbol{x};\boldsymbol{x}_0)$. The closed forms of $u^*_{\alpha\beta}(\boldsymbol{x};\boldsymbol{x}_0)$ and $p^*_{\alpha\beta}(\boldsymbol{x};\boldsymbol{x}_0)$ are given by

$$
\begin{aligned}
u^*_{\alpha\beta} &= \frac{1}{8\pi\mu(1-\nu)}\left\{(3-4\nu)\log\left(\frac{1}{r}\right)\delta_{\alpha\beta}+\frac{r_\alpha r_\beta}{r^2}\right\},\\
p^*_{\alpha\beta} &= \frac{-1}{4\pi(1-\nu)r}\left[\frac{\partial r}{\partial n}\left\{(1-2\nu)\delta_{\alpha\beta}+\frac{2r_\alpha r_\beta}{r^2}\right\}-(1-2\nu)\frac{n_\alpha r_\beta - n_\beta r_\alpha}{r}\right],
\end{aligned}
$$

where $\boldsymbol{r}\equiv\boldsymbol{x}-\boldsymbol{x}_0$, $r\equiv|\boldsymbol{r}|$ and $\partial/\partial n$ stands for the normal derivative $\boldsymbol{x}\in\Gamma$. Besides, ν is the Poisson's ratio given by $2\nu=\lambda/(\lambda+\mu)$. Then, (1) and Gauss's theorem give rise to the boundary integral equation

$$
\frac{1}{2}u_\alpha(\boldsymbol{x})+\sum_\beta\int_\Gamma p^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x})u_\beta(\boldsymbol{y})\mathrm{d}\Gamma(\boldsymbol{y})=\sum_\beta\int_\Gamma u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x})p_\beta(\boldsymbol{y})\mathrm{d}\Gamma(\boldsymbol{y}),\quad \boldsymbol{x}\in\Gamma. \tag{5}
$$

Next, the discretization of (5) is considered. First, the boundary Γ is approximated by N line segments, i.e.

$$
\Gamma=\sum_{i=1}^{N}\Gamma_i,
$$

where $\sum$ represents the union of sets without any intersection except for their boundaries. Each Γ_i is called a boundary element. Let $\boldsymbol{x}_i$ be the representative point of Γ_i, e.g. its mid point. Also assume that $u_\alpha(\boldsymbol{x})$ and $p_\alpha(\boldsymbol{x})$ remain constant within each boundary element Γ_i, i.e.

$$
u_{i\alpha}\equiv u_\alpha(\boldsymbol{x}),\quad p_{i\alpha}\equiv p_\alpha(\boldsymbol{x}),\qquad \boldsymbol{x}\in\Gamma_i.
$$

Then, (5) can be discretized as

$$
\frac{1}{2}u_{i\alpha}+\sum_{j,\beta}u_{j\beta}\int_{\Gamma_j}p^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x}_i)\mathrm{d}\Gamma(\boldsymbol{y}) = \sum_{j,\beta}p_{j\beta}\int_{\Gamma_j}u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x}_i)\mathrm{d}\Gamma(\boldsymbol{y}). \tag{6}
$$

Let

$$
\begin{aligned}
g_{i\alpha,j\beta} &\equiv \int_{\Gamma_j}u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x}_i)\mathrm{d}\Gamma(\boldsymbol{y}),\\
h_{i\alpha,j\beta} &\equiv \int_{\Gamma_j}p^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x}_i)\mathrm{d}\Gamma(\boldsymbol{y})+\frac{1}{2}\delta_{\alpha\beta}\delta_{ij},\\
G &\equiv (g_{i\alpha,j\beta}),\qquad H\equiv(h_{i\alpha,j\beta}),\\
\boldsymbol{U} &\equiv (u_{11}\ u_{12}\ u_{21}\ \cdots u_{N2})^{\mathrm{T}},\quad \boldsymbol{P}\equiv(p_{11}\ p_{12}\ p_{21}\ \cdots p_{N2})^{\mathrm{T}}.
\end{aligned}
$$

Using these notations, (6) can be expressed in matrix vector form as

$$
H\boldsymbol{U}=G\boldsymbol{P}. \tag{7}
$$

Let

$$
\boldsymbol{U}\equiv\begin{pmatrix}\boldsymbol{U}^{(\mathrm{D})}\\ \boldsymbol{U}^{(\mathrm{N})}\end{pmatrix},\qquad \boldsymbol{P}\equiv\begin{pmatrix}\boldsymbol{P}^{(\mathrm{D})}\\ \boldsymbol{P}^{(\mathrm{N})}\end{pmatrix},
$$

where the entries of $\boldsymbol{U}^{(\mathrm{D})}$ and $\boldsymbol{P}^{(\mathrm{D})}$ are the values of $u_{i\alpha}$ and $p_{i\alpha}$, respectively, on $\Gamma^{(\mathrm{D})}$, and the entries of $\boldsymbol{U}^{(\mathrm{N})}$ and $\boldsymbol{P}^{(\mathrm{N})}$ are the values on $\Gamma^{(\mathrm{N})}$. Then $\boldsymbol{U}^{(\mathrm{D})}$ and $\boldsymbol{P}^{(\mathrm{N})}$ are known values, and $\boldsymbol{U}^{(\mathrm{N})}$ and $\boldsymbol{P}^{(\mathrm{D})}$ are unknown values. By bringing the unknowns to the left hand side, equation(7) yields a system of linear equations

$$\left(\begin{array}{cc} -G^{(\mathrm{D})} & H^{(\mathrm{N})} \end{array} \right) \left(\begin{array}{c} \boldsymbol{P}^{(\mathrm{D})} \\ \boldsymbol{U}^{(\mathrm{N})} \end{array} \right) = \left(\begin{array}{cc} -H^{(\mathrm{D})} & G^{(\mathrm{N})} \end{array} \right) \left(\begin{array}{c} \boldsymbol{U}^{(\mathrm{D})} \\ \boldsymbol{P}^{(\mathrm{N})} \end{array} \right), \tag{8}$$

where $\left(-G^{(\mathrm{D})}\ H^{(\mathrm{N})}\right)$ is a dense non-symmetric matrix.

3. The multipole method.

3.1. The basic idea of the multipole method. In the standard boundary element method, (8) is solved using LU decomposition, requiring $\mathrm{O}\left(N^3\right)$ computation time and $\mathrm{O}\left(N^2\right)$ memory.

If one were to apply iterative methods such as GMRES [5] or Bi-CGSTAB [6] for the non-symmetric system of linear equations, computation time may be reduced provided the number of iterations is less than $\mathrm{O}(N)$. However, memory and computation time of $\mathrm{O}\left(N^2\right)$ is still required to perform the dense matrix vector multiplication for each iteration. Hence, one would like to reduce this bottleneck.

Let us, for the moment, consider the Dirichlet problem, where $\boldsymbol{P}$ is to be solved for given $\boldsymbol{U}$ in equation (7). Then, in order to apply an iterative solver, one must be able to evaluate the inner product between any row vector of G and an iteration vector

$$\boldsymbol{q} \equiv \left(q_{11}\ q_{12}\ q_{21}\ \cdots\ q_{N2}\right)^{\mathrm{T}}$$

to compute $G\boldsymbol{q}$.

From (6), the inner product related to the observation point $\boldsymbol{x}_i$ is given by

$$I_{i\alpha} \equiv \sum_{j=1}^{N} \int_{\Gamma_j} \sum_{\beta} u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x}_i) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}).$$

Define $\Gamma_i^{(\mathrm{n})}$ as the set of boundary elements Γ_j which are near $\boldsymbol{x}_i$. Assume that the size of the set $\Gamma_i^{(\mathrm{n})}$ is a constant independent of N, and let

$$\Gamma_i^{(\mathrm{f})} \equiv \Gamma - \Gamma_i^{(\mathrm{n})}.$$

To distinguish between the contributions of $\Gamma_i^{(\mathrm{n})}$ and $\Gamma_i^{(\mathrm{f})}$ to $I_{i\alpha}$, denote $I_{i\alpha}$ as

$$\begin{aligned} I_{i\alpha} &= \sum_{\beta} I^{(\mathrm{n})}_{i\alpha\beta} + \sum_{\beta} I^{(\mathrm{f})}_{i\alpha\beta}, \\ I^{(\mathrm{n})}_{i\alpha\beta} &\equiv \int_{\Gamma_i^{(\mathrm{n})}} u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x}_i) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}), \\ I^{(\mathrm{f})}_{i\alpha\beta} &\equiv \int_{\Gamma_i^{(\mathrm{f})}} u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x}_i) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}). \end{aligned}$$

First, the near contribution $I^{(\mathrm{n})}_{i\alpha}$ is calculated directly using

$$I^{(\mathrm{n})}_{i\alpha\beta} = \sum_{\Gamma_j \subset \Gamma_i^{(\mathrm{n})}} g_{i\alpha,j\beta} q_{j\beta},$$

where

$$g_{i\alpha,j\beta} = \int_{\Gamma_j} u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x}_i)\mathrm{d}\Gamma(\boldsymbol{y}).$$

Next, consider the calculation of the far contribution $I^{(\mathrm{f})}_{i\alpha\beta}$. Construct clusters G_h by assembling boundary elements Γ_j which are included in $\Gamma^{(\mathrm{f})}_i$. G_h satisfy

$$G_h \equiv \sum_j \Gamma_j\,, \qquad \Gamma^{(\mathrm{f})}_i = \sum_h G_h\,, \tag{9}$$

i.e. G_h is a disjoint union of boundary elements Γ_j, and $\Gamma^{(\mathrm{f})}_i$ in turn is a disjoint union of clusters G_h. The clusters must be formed so that there exists $\boldsymbol{C}_h$ for each cluster G_h such that

$$|\boldsymbol{x}_i - \boldsymbol{C}_h| > 2\,|\boldsymbol{y} - \boldsymbol{C}_h|, \qquad \forall \boldsymbol{y} \in G_h\,,$$

as shown in Fig. 1. $\boldsymbol{C}_h$ is called the center of cluster G_h.

In order to reduce memory and computation time, the G_h's are formed so that they can be shared among different $\boldsymbol{x}_i$'s, and so that the total number of clusters G_h is $\mathrm{O}(N)$. Once the clusters G_h are formed, one chooses which clusters G_h are required to form $\Gamma^{(\mathrm{f})}_i$ before calculating each $I_{i\alpha\beta}$. Let H_i be the set of indices h of G_h which are required to form $\Gamma^{(\mathrm{f})}_i$. We will refer to H_i as the cluster list for $\boldsymbol{x}_i$. Then, (9) can be written more precisely as

$$\Gamma^{(\mathrm{f})}_i = \sum_{h\in H_i} G_h\,.$$

More specifically, the clusters G_h are formed by recursively halving the rectangular

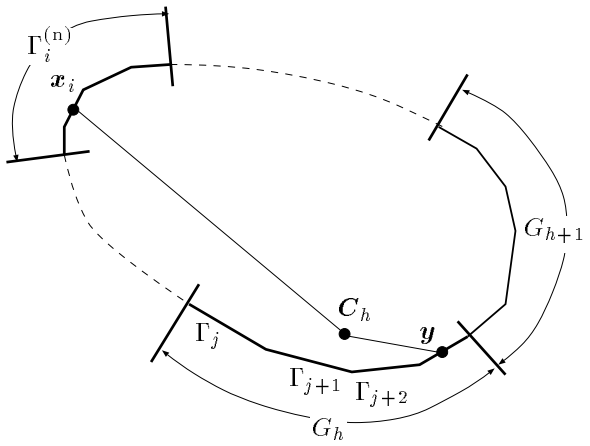

FIG. 1. *Observation point, source point and cluster.*

region containing Ω in alternate directions and assigning a cluster G_h to the set of elements contained in the sub-rectangle, as shown in Fig. 2.

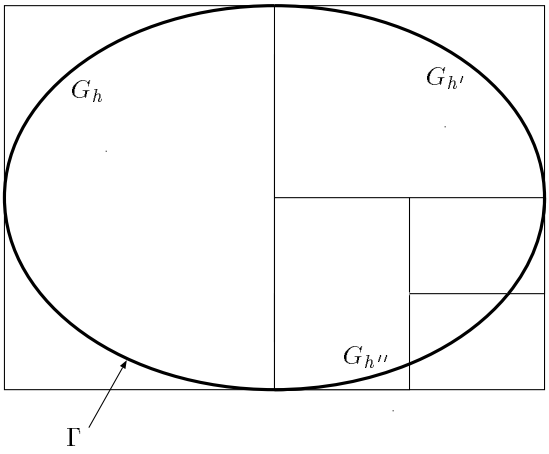

FIG. 2. *Formation of clusters* G_h.

Let $\boldsymbol{r} \equiv \boldsymbol{y} - \boldsymbol{x}_i$, $r \equiv |\boldsymbol{r}|$. If we can expand each term of equation

$$
\begin{aligned}
I^{(\mathrm{f})}_{i\alpha\beta} &= \sum_h \int_{G_h} u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x}_i) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}) \\
&= \sum_h \frac{1}{8\pi\mu(1-\nu)} \int_{G_h} \left\{ (3-4\nu) \log\left(\frac{1}{r}\right) \delta_{\alpha\beta} + \frac{r_\alpha r_\beta}{r^2} \right\} q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}) \qquad (10)
\end{aligned}
$$

into Taylor series around $\boldsymbol{C}_h$, and truncate them at the p-th term to obtain

$$
\sum_h \sum_{k=0}^{p} F(\boldsymbol{x}_i - \boldsymbol{C}_h; k)\, c(G_h, k), \qquad (11)
$$

where $c(G_h, k)$ is independent of $\boldsymbol{x}_i$, and $p \approx \log(\delta)$, where δ is the required accuracy in computing the inner product, then $c(G_h, k)$ can be used to calculate inner products for many $\boldsymbol{x}_i$'s, thus reducing the total computation time and memory in computing the matrix vector product $G\boldsymbol{q}$.

3.2. Multipole expansion of the two dimensional elastostatic kernel. First, consider the expansion of the term

$$
\int_{G_h} \log(r) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}),
$$

in the first term of (10). For convenience, identify $\mathbf{R}^2$ with the complex plain $\mathbf{C}$ and regard $\boldsymbol{x}_i, \boldsymbol{y}, \boldsymbol{C}_h \in \mathbf{C}$. In other words, regard $\boldsymbol{x} \equiv (x_1\ x_2) \in \mathbf{R}^2$ as $\boldsymbol{x} = x_1 + \mathrm{i}x_2 \in \mathbf{C}$ and so on. Then,

$$
\begin{aligned}
\int_{G_h} \log(r) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}) &= \mathrm{Re} \int_{G_h} \log(\boldsymbol{x}_i - \boldsymbol{y}) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}) \\
&= \mathrm{Re} \int_{G_h} \left\{ \log(\boldsymbol{x}_i - \boldsymbol{C}_h) + \sum_{k>0} \frac{-1}{k} \frac{(\boldsymbol{y} - \boldsymbol{C}_h)^k}{(\boldsymbol{x}_i - \boldsymbol{C}_h)^k} \right\} q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}),
\end{aligned}
$$

so that

$$\int_{G_h} \log(r) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}) = \mathrm{Re}\left[\sum_k f(\boldsymbol{x}_i - \boldsymbol{C}_h; k)\, A_\beta(G_h, k)\right], \tag{12}$$

where

$$\begin{aligned} a_\beta(G_h, k) &\equiv \int_{G_h} (\boldsymbol{y} - \boldsymbol{C}_h)^k q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}), \\ f(\boldsymbol{x}; k) &\equiv \begin{cases} \log(\boldsymbol{x}), & k = 0, \\ \dfrac{1}{\boldsymbol{x}^k}, & k = 1, 2, \ldots, \end{cases} \end{aligned}$$

and

$$A_\beta(G_h, k) \equiv \begin{cases} a_\beta(G_h, 0), & k = 0, \\ -\dfrac{1}{k} a_\beta(G_h, k), & k = 1, 2, \ldots \end{cases}.$$

Hence, the form of (11) is obtained by truncating at the p-th term.

Next, consider the second term of (10) with $\alpha = \beta = 1$. Still regarding $\boldsymbol{r} = r_1 + \mathrm{i} r_2$, the equation

$$\frac{r_1}{r^2} = \mathrm{Re}\frac{1}{\boldsymbol{r}} = -\mathrm{Re}\frac{1}{(\boldsymbol{x}_i - \boldsymbol{C}_h) - (\boldsymbol{y} - \boldsymbol{C}_h)} = -\mathrm{Re}\sum_{k \geq 0} \frac{(\boldsymbol{y} - \boldsymbol{C}_h)^k}{(\boldsymbol{x}_i - \boldsymbol{C}_h)^{k+1}}$$

leads to the expansion

$$\begin{aligned} \frac{r_1^2}{r^2} &= -\mathrm{Re}\sum_{k \geq 0} \frac{(\boldsymbol{y} - \boldsymbol{C}_h)^k}{(\boldsymbol{x}_i - \boldsymbol{C}_h)^{k+1}} \mathrm{Re}\boldsymbol{r} \\ &= \mathrm{Re}\left\{\sum_{k \geq 0} \frac{(\boldsymbol{y} - \boldsymbol{C}_h)^k}{(\boldsymbol{x}_i - \boldsymbol{C}_h)^{k+1}}\right\} \mathrm{Re}(\boldsymbol{x}_i - \boldsymbol{C}_h) \\ &\quad -\mathrm{Re}\left\{\sum_{k \geq 0} \frac{(\boldsymbol{y} - \boldsymbol{C}_h)^k}{(\boldsymbol{x}_i - \boldsymbol{C}_h)^{k+1}}\right\} \mathrm{Re}(\boldsymbol{y} - \boldsymbol{C}_h). \end{aligned}$$

Besides, since $\mathrm{Re}\, z_1 \,\mathrm{Re}\, z_2 = \mathrm{Re}\,(z_1 \mathrm{Re}\, z_2)$ for arbitrary complex numbers z_1 and z_2,

$$\begin{aligned} \int_{G_h} \frac{r_1^2}{r^2} q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}) &= \mathrm{Re}\left[\int_{G_h} \sum_{k \geq 0} \frac{(\boldsymbol{y} - \boldsymbol{C}_h)^k}{(\boldsymbol{x}_i - \boldsymbol{C}_h)^{k+1}} \mathrm{Re}(\boldsymbol{x}_i - \boldsymbol{C}_h) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y})\right] \\ &\quad -\mathrm{Re}\left[\int_{G_h} \sum_{k \geq 0} \frac{(\boldsymbol{y} - \boldsymbol{C}_h)^k}{(\boldsymbol{x}_i - \boldsymbol{C}_h)^{k+1}} \mathrm{Re}(\boldsymbol{y} - \boldsymbol{C}_h) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y})\right]. \end{aligned}$$

If we now let

$$\begin{aligned} a_\beta^{(\mathrm{r})}(G_h, k) &\equiv \int_{G_h} (\boldsymbol{y} - \boldsymbol{C}_h)^k \mathrm{Re}(\boldsymbol{y} - \boldsymbol{C}_h) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}), \\ a_\beta^{(\mathrm{i})}(G_h, k) &\equiv \int_{G_h} (\boldsymbol{y} - \boldsymbol{C}_h)^k \mathrm{Im}(\boldsymbol{y} - \boldsymbol{C}_h) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}), \\ f^{(\mathrm{r})}(\boldsymbol{x}; k) &\equiv \frac{\mathrm{Re}\,\boldsymbol{x}}{\boldsymbol{x}^k}, \\ f^{(\mathrm{i})}(\boldsymbol{x}; k) &\equiv \frac{\mathrm{Im}\,\boldsymbol{x}}{\boldsymbol{x}^k}, \end{aligned}$$

we obtain

$$\int_{G_h} \frac{r_1^2}{r^2} q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}) = \mathrm{Re}\left[\sum_k f^{(\mathrm{r})}(\boldsymbol{x}_i - \boldsymbol{C}_h; k+1) a_\beta(G_h, k)\right] - \mathrm{Re}\left[\sum_k f(\boldsymbol{x}_i - \boldsymbol{C}_h; k+1) a_\beta^{(\mathrm{r})}(G_h, k)\right]. \tag{13}$$

Hence, the second term of (10) can also be expressed in the form of (11). From (10), (12) and (13), we have

$$\begin{aligned} 8\pi\mu(1-\nu) I_{i11}^{(\mathrm{f})} &= (4\nu - 3)\mathrm{Re}\left[\sum_{h,k} f(\boldsymbol{x}_i - \boldsymbol{C}_h; k) A_1(G_h, k)\right] \\ &\quad + \mathrm{Re}\left[\sum_{h,k} f^{(\mathrm{r})}(\boldsymbol{x}_i - \boldsymbol{C}_h; k+1) a_1(G_h, k)\right] \\ &\quad - \mathrm{Re}\left[\sum_{h,k} f(\boldsymbol{x}_i - \boldsymbol{C}_h; k+1) a_1^{(\mathrm{r})}(G_h, k)\right], \end{aligned} \tag{14}$$

where $a_\beta(G_h, k)$, $a_\beta^{(\mathrm{r})}(G_h, k)$ can be shared by many $\boldsymbol{x}_i$'s.

Similar results follow for $\alpha = 1$, $\beta = 2$ and $\alpha = 2$ by replacing Re by Im.

3.3. The procedure for computing the inner product. In order to evaluate the inner products which constitute the matrix vector product in the iterative solver, we take the following procedure. First, the values of $a_\beta(G_h, k)$, $a_\beta^{(\mathrm{r})}(G_h, k)$ and $a_\beta^{(\mathrm{i})}(G_h, k)$ are computed for all the clusters G_h. The amount of computation for this step is $\mathrm{O}(N \log N)$, since the time to compute $a_\beta(G_h, k)$ $a_\beta^{(\mathrm{r})}(G_h, k)$ and $a_\beta^{(\mathrm{i})}(G_h, k)$ for all the clusters G_h at any level l is proportional to N, and the number of levels of clusters is $\mathrm{O}(N)$. Fig. 3 shows a one dimensional schematic diagram.

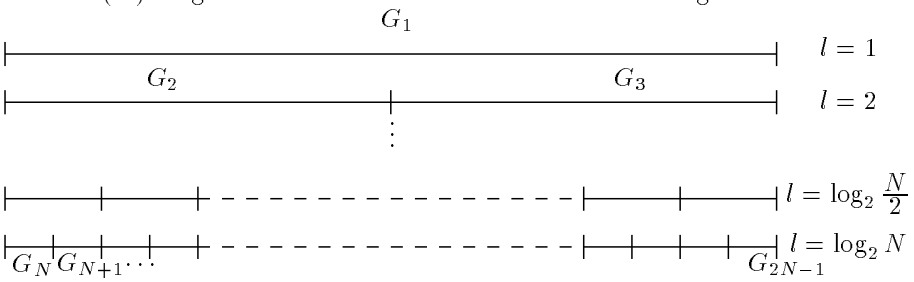

FIG. 3. *Clusters G_h at level l.*

Next, the inner product for each observation point $\boldsymbol{x}_i$ are computed using (14). The cost of computing an inner product for one observation point $\boldsymbol{x}_i$ is $\mathrm{O}(\log N)$, since an $\mathrm{O}(1)$ number of clusters are required from each level. Hence, the total computation for the inner products for all the points $\boldsymbol{x}_i$, $i = 1, 2, \ldots N$ is also $\mathrm{O}(N \log N)$.

Thus, we can conclude that the amount of computation to solve the original system of linear equations (6) by an iterative method is $\mathrm{O}(N \log N)$ per one iteration.

(One may improve the order to $\mathrm{O}(N)$ using techniques similar to those in [3].) Hence, if M iterations are required to solve equation (6), the total amount of computation is $\mathrm{O}(MN\log N)$.

If we adopt the Bi-CGSTAB method [6] as the iterative method, all we have to save are $a_\beta(G_h,k)$, $a_\beta^{(\mathrm{r})}(G_h,k)$, $a_\beta^{(\mathrm{i})}(G_h,k)$ and some $2N$-dimensional vectors, all of size $\mathrm{O}(N)$. This reduces the memory requirement to $\mathrm{O}(N\log N)$. Note that the memory required is $\mathrm{O}(N)$ for storing $a_\beta(G_h,k)$, $a_\beta^{(\mathrm{r})}(G_h,k)$, $a_\beta^{(\mathrm{i})}(G_h,k)$ (floating numbers), but the memory required for storing the integer list of clusters related to each observation point $\boldsymbol{x}_i$ is $\mathrm{O}(N\log N)$. However, the number of iterations M may become large if the problem is ill-conditioned. On the other hand, if we adopt the GMRES method [5], it is guaranteed that M is not larger than the number of unknowns $2N$, in theory. But the memory requirement is $\mathrm{O}(MN)$, since we must save all the Krylov bases. Then, the memory requirement increases with M.

3.4. The computation of the known vector for the Dirichlet problem. In the standard technique, the computation of the right hand side $H\boldsymbol{U}$ of (6), or the known vector, also requires $\mathrm{O}(N^2)$ operations, although the memory requirement is $\mathrm{O}(N)$. The multipole method can also be applied to this phase.

Let the inner product between the row vector of H and the temporary vector $\boldsymbol{q}$ related to the observation point $\boldsymbol{x}_i$ be $J_{i\alpha\beta}$, and let the part of $J_{i\alpha\beta}$ due $\Gamma_i^{(\mathrm{f})}$ be $J_{i\alpha\beta}^{(\mathrm{f})}$. Then, we have

$$
\begin{aligned}
J_{i\alpha\beta}^{(\mathrm{f})} &= \sum_h \int_{G_h} p_{\alpha\beta}^*(\boldsymbol{y};\boldsymbol{x}_i) q_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y}) \\
&= \sum_h \int_{G_h} \frac{-1}{4\pi(1-\nu)r}\left[\frac{\partial r}{\partial n}\left\{(1-2\nu)\delta_{\alpha\beta}+\frac{2r_\alpha r_\beta}{r^2}\right\}+(1-2\nu)\frac{n_\alpha r_\beta - n_\beta r_\alpha}{r}\right]\mathrm{d}\Gamma(\boldsymbol{y}).
\end{aligned}
$$

Now consider the Taylor expansion of each term of $J_{i\alpha\beta}^{(\mathrm{f})}$ around $\boldsymbol{C}_h$. First, since

$$
\frac{\boldsymbol{n}}{\boldsymbol{r}} = \frac{1}{r}\frac{\partial r}{\partial n} + \mathrm{i}\frac{n_2 r_1 - n_1 r_2}{r^2},
$$

$\frac{1}{r}\frac{\partial r}{\partial n}$ and $(n_2 r_1 - n_1 r_2)/r^2$ can be expanded as

$$
\begin{aligned}
\frac{1}{r}\frac{\partial r}{\partial n} &= \mathrm{Re}\frac{\boldsymbol{n}}{\boldsymbol{r}} = \mathrm{Re}\left[\boldsymbol{n}\sum_{k\geq 0}\frac{(\boldsymbol{y}-\boldsymbol{C}_h)^k}{(\boldsymbol{x}_i-\boldsymbol{C}_h)^{k+1}}\right], \\
\frac{n_2 r_1 - n_1 r_2}{r^2} &= \mathrm{Im}\frac{\boldsymbol{n}}{\boldsymbol{r}} = \mathrm{Im}\left[\boldsymbol{n}\sum_{k\geq 0}\frac{(\boldsymbol{y}-\boldsymbol{C}_h)^k}{(\boldsymbol{x}_i-\boldsymbol{C}_h)^{k+1}}\right].
\end{aligned}
$$

Hence, we obtain

$$
\begin{aligned}
\int_{G_h}\frac{1}{r}\frac{\partial r}{\partial n} q_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y}) &= \mathrm{Re}\left[\int_{G_h}\boldsymbol{n}\sum_{k\geq 0}\frac{(\boldsymbol{y}-\boldsymbol{C}_h)^k}{(\boldsymbol{x}_i-\boldsymbol{C}_h)^{k+1}} q_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y})\right], \\
\int_{G_h}\frac{n_2 r_1 - n_1 r_2}{r^2} q_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y}) &= \mathrm{Im}\left[\int_{G_h}\boldsymbol{n}\sum_{k\geq 0}\frac{(\boldsymbol{y}-\boldsymbol{C}_h)^k}{(\boldsymbol{x}_i-\boldsymbol{C}_h)^{k+1}} q_{j\beta},\mathrm{d}\Gamma(\boldsymbol{y})\right],
\end{aligned}
$$

or letting

$$b_\beta(G_h,k) \equiv \int_{G_h} \boldsymbol{n}(\boldsymbol{y}-\boldsymbol{C}_h)^k q_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y}),$$

we have

$$\begin{aligned}\int_{G_h}\frac{1}{r}\frac{\partial r}{\partial n}\mathrm{d}\Gamma(\boldsymbol{y}) &= \mathrm{Re}\left[\sum_{k\geq 0} f(\boldsymbol{x}_i-\boldsymbol{C}_h;k)b(G_h,k+1)\right],\\ \int_{G_h}\frac{n_2r_1-n_1r_2}{r^2}\mathrm{d}\Gamma(\boldsymbol{y}) &= \mathrm{Im}\left[\sum_{k\geq 0} f(\boldsymbol{x}_i-\boldsymbol{C}_h;k)b(G_h,k+1)\right].\end{aligned}$$

Next, the equation

$$\begin{aligned}\frac{1}{r}\frac{\partial r}{\partial n}\frac{2r_1r_2}{r^2} &= -\mathrm{Re}\,(\boldsymbol{r}\bar{\boldsymbol{n}})\,\mathrm{Im}\left(\frac{1}{\boldsymbol{r}}\right)^2\\ &= \mathrm{Re}\,(\boldsymbol{r}\bar{\boldsymbol{n}})\,\mathrm{Im}\left\{\sum_{k\geq 0}(k+1)\frac{(\boldsymbol{y}-\boldsymbol{C}_h)^k}{(\boldsymbol{x}_i-\boldsymbol{C}_h)^{k+2}}\right\}\end{aligned}$$

leads to

$$\begin{aligned}&\int_{G_h}\frac{1}{r}\frac{\partial r}{\partial n}\frac{2r_1r_2}{r^2}\mathrm{d}\Gamma(\boldsymbol{y})\\ =\ &\mathrm{Im}\left[\int_{G_h}\mathrm{Re}\,\{(\boldsymbol{y}-\boldsymbol{C}_h)\bar{\boldsymbol{n}}\}\sum_{k\geq 0}(k+1)\frac{(\boldsymbol{y}-\boldsymbol{C}_h)^k}{(\boldsymbol{x}_i-\boldsymbol{C}_h)^{k+2}}\mathrm{d}\Gamma(\boldsymbol{y})\right]\\ &-\mathrm{Im}\left[\int_{G_h}\mathrm{Re}\,\{(\boldsymbol{x}_i-\boldsymbol{C}_h)\bar{\boldsymbol{n}}\}\sum_{k\geq 0}(k+1)\frac{(\boldsymbol{y}-\boldsymbol{C}_h)^k}{(\boldsymbol{x}_i-\boldsymbol{C}_h)^{k+2}}\mathrm{d}\Gamma(\boldsymbol{y})\right],\end{aligned}$$

where $\bar{\boldsymbol{n}}$ denotes the complex conjugate of $\boldsymbol{n}$. If we define

$$\begin{aligned}b_\beta^{(\mathrm{r})}(G_h,k) &\equiv \int_{G_h}\mathrm{Re}(\boldsymbol{n})(\boldsymbol{y}-\boldsymbol{C}_h)^k q_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y}),\\ b_\beta^{(\mathrm{i})}(G_h,k) &\equiv \int_{G_h}\mathrm{Im}(\boldsymbol{n})(\boldsymbol{y}-\boldsymbol{C}_h)^k q_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y}),\\ b_\beta^{(\mathrm{rr})}(G_h,k) &\equiv \int_{G_h}\mathrm{Re}(\boldsymbol{y}-\boldsymbol{C}_h)\mathrm{Re}(\boldsymbol{n})(\boldsymbol{y}-\boldsymbol{C}_h)^k q_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y}),\\ b_\beta^{(\mathrm{ii})}(G_h,k) &\equiv \int_{G_h}\mathrm{Im}(\boldsymbol{y}-\boldsymbol{C}_h)\mathrm{Im}(\boldsymbol{n})(\boldsymbol{y}-\boldsymbol{C}_h)^k q_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y}),\end{aligned}$$

we have

$$\begin{aligned}&\int_{G_h}\frac{1}{r}\frac{\partial r}{\partial n}\frac{2r_1r_2}{r^2}\mathrm{d}\Gamma(\boldsymbol{y})\\ =\ &\mathrm{Im}\left[\sum_{k\geq 0}\left\{b_\beta^{(\mathrm{rr})}(G_h,k)+b_\beta^{(\mathrm{ii})}(G_h,k)\right\}f(\boldsymbol{x}_i-\boldsymbol{C}_h;k+2)\right]\\ &-\mathrm{Im}\left[\sum_{k\geq 0}\left\{b_\beta^{(\mathrm{r})}(G_h,k)f^{(\mathrm{r})}(\boldsymbol{x}_i-\boldsymbol{C}_h;k+2)+b_\beta^{(\mathrm{i})}(G_h,k)f^{(\mathrm{i})}(\boldsymbol{x}_i-\boldsymbol{C}_h;k+2)\right\}\right].\end{aligned}$$

Finally, we are left with the expansion of

$$\frac{1}{r}\frac{\partial r}{\partial n}\frac{r_\alpha r_\alpha}{r^2}.$$

Instead of expanding this term, we expand the term

$$\frac{1}{r}\frac{\partial r}{\partial n}\frac{r_1^2-r_2^2}{r^2}.$$

This is sufficient, since the expansion of

$$\frac{1}{r}\frac{\partial r}{\partial n}\frac{r_1^2+r_2^2}{r^2}=\frac{1}{r}\frac{\partial r}{\partial n}$$

has already been obtained. Since

$$\begin{aligned}\frac{1}{r}\frac{\partial r}{\partial n}\frac{r_1^2-r_2^2}{r^2} &= \mathrm{Re}\,(\boldsymbol{rn})\,\mathrm{Re}\left(\frac{1}{\boldsymbol{r}}\right)^2\\ &= \mathrm{Re}\,(\boldsymbol{rn})\,\mathrm{Re}\left\{\sum_{k\geq 0}(k+1)\frac{(\boldsymbol{y}-\boldsymbol{C}_h)^k}{(\boldsymbol{x}_i-\boldsymbol{C}_h)^{k+2}}\right\},\end{aligned}$$

we obtain

$$\begin{aligned}&\int_{G_h}\frac{1}{r}\frac{\partial r}{\partial n}\frac{r_1^2-r_2^2}{r^2}\mathrm{d}\Gamma(\boldsymbol{y})\\ =\;&\mathrm{Re}\left[\int_{G_h}\mathrm{Re}\,\{(\boldsymbol{y}-\boldsymbol{C}_h)\boldsymbol{n}\}\sum_{k\geq 0}(k+1)\frac{(\boldsymbol{y}-\boldsymbol{C}_h)^k}{(\boldsymbol{x}_i-\boldsymbol{C}_h)^{k+2}}\mathrm{d}\Gamma(\boldsymbol{y})\right]\\ &-\mathrm{Re}\left[\int_{G_h}\mathrm{Re}\,\{(\boldsymbol{x}_i-\boldsymbol{C}_h)\boldsymbol{n}\}\sum_{k\geq 0}(k+1)\frac{(\boldsymbol{y}-\boldsymbol{C}_h)^k}{(\boldsymbol{x}_i-\boldsymbol{C}_h)^{k+2}}\mathrm{d}\Gamma(\boldsymbol{y})\right]\\ =\;&\mathrm{Re}\left[\sum_{k\geq 0}(k+1)\left\{b_\beta^{(\mathrm{rr})}(G_h,k)+b_\beta^{(\mathrm{ii})}(G_h,k)\right\}f(\boldsymbol{x}_i-\boldsymbol{C}_h;k+2)\right]\\ &-\mathrm{Re}\left[\sum_{k\geq 0}\left\{b_\beta^{(\mathrm{r})}(G_h,k-1)f^{(\mathrm{r})}(\boldsymbol{x}_i-\boldsymbol{C}_h;k+2)\right.\right.\\ &\qquad\left.\left.+b_\beta^{(\mathrm{i})}(G_h,k-1)f^{(\mathrm{i})}(\boldsymbol{x}_i-\boldsymbol{C}_h;k+2)\right\}\right].\end{aligned}$$

By the above argument, equations such as

$$\begin{aligned}&-4\pi(1-\nu)J_{i11}^{(\mathrm{f})}\\ =\;&2(1-\nu)\mathrm{Re}\left[\sum_{k\geq 0}f(\boldsymbol{x}_i-\boldsymbol{C}_h;k+1)b_\beta(G_h,k)\right]\\ &+\mathrm{Re}\left[\sum_{k\geq 0}(k+1)\left\{b_\beta^{(\mathrm{rr})}(G_h,k)+b_\beta^{(\mathrm{ii})}(G_h,k)\right\}f(\boldsymbol{x}_i-\boldsymbol{C}_h;k+2)\right]\end{aligned}$$

$$-\mathrm{Re}\left[\sum_{k\geq 0}(k+1)\left\{b_\beta^{(\mathrm{r})}(G_h,k-1)f^{(\mathrm{r})}(\boldsymbol{x}_i-\boldsymbol{C}_h;k+2)\right.\right.$$
$$\left.\left.+b_\beta^{(\mathrm{i})}(G_h,k-1)f^{(\mathrm{i})}(\boldsymbol{x}_i-\boldsymbol{C}_h;k+2)\right\}\right]$$

are obtained.

So far, we have only treated the Dirichlet problem, but we can similarly treat the Neumann problem or the mixed boundary value problem, by exchanging the role of the known and the unknown variables.

3.5. Calculation of the interior displacement and stress. Once all the quantities $u_{j\beta}$ and $p_{j\beta}$ on the boundary Γ are determined by the above procedures, the displacement $\boldsymbol{u}(\boldsymbol{x})$ at any interior point $\boldsymbol{x}$ of the region Ω can be computed using

$$u_\alpha(\boldsymbol{x}) = -\sum_{j,\beta} u_{j\beta}\int_{\Gamma_j} p^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x})\mathrm{d}\Gamma(\boldsymbol{y}) + \sum_{j,\beta} p_{j\beta}\int_{\Gamma_j} u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x})\mathrm{d}\Gamma(\boldsymbol{y}). \tag{15}$$

If we define

$$I_{\alpha\beta}(\boldsymbol{x}) \equiv \sum_j p_{j\beta}\int_{\Gamma_j} u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x})\mathrm{d}\Gamma(\boldsymbol{y}), \qquad J_{\alpha\beta}(\boldsymbol{x}) \equiv \sum_j u_{j\beta}\int_{\Gamma_j} p^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x})\mathrm{d}\Gamma(\boldsymbol{y}),$$

(15) becomes

$$u_\alpha(\boldsymbol{x}) = \sum_\beta \left\{I_{\alpha\beta}(\boldsymbol{x}) - J_{\alpha\beta}(\boldsymbol{x})\right\}. \tag{16}$$

In the following, we show how to compute $I_{\alpha\beta}(\boldsymbol{x})$ and $J_{\alpha\beta}(\boldsymbol{x})$ using the multipole method. Similar to sections 3.1 and 3.4, let $\Gamma^{(\mathrm{n})}(\boldsymbol{x})$ be the set of boundary elements Γ_j which are near $\boldsymbol{x}$, and let $\Gamma^{(\mathrm{f})}(\boldsymbol{x}) \equiv \Gamma - \Gamma^{(\mathrm{n})}(\boldsymbol{x})$. To distinguish the contributions of $\Gamma^{(\mathrm{n})}(\boldsymbol{x})$ and $\Gamma^{(\mathrm{f})}(\boldsymbol{x})$ to $I_{\alpha\beta}(\boldsymbol{x})$, let

$$\begin{aligned} I_{\alpha\beta}(\boldsymbol{x}) &= I^{(\mathrm{n})}_{\alpha\beta}(\boldsymbol{x}) + I^{(\mathrm{f})}_{\alpha\beta}(\boldsymbol{x}), \\ I^{(\mathrm{n})}_{\alpha\beta}(\boldsymbol{x}) &\equiv \int_{\Gamma^{(\mathrm{n})}(\boldsymbol{x})} u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x})p_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y}), \\ I^{(\mathrm{f})}_{\alpha\beta}(\boldsymbol{x}) &\equiv \int_{\Gamma^{(\mathrm{f})}(\boldsymbol{x})} u^*_{\alpha\beta}(\boldsymbol{y};\boldsymbol{x})p_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y}). \end{aligned}$$

Similar to the calculation of $I^{(\mathrm{f})}_{i\alpha\beta}$ of (14), the Taylor expansion of $I^{(\mathrm{f})}_{\alpha\beta}(\boldsymbol{x})$ around $\boldsymbol{C}_h$ leads to the equations

$$\begin{aligned} 8\pi\mu(1-\nu)I^{(\mathrm{f})}_{11}(\boldsymbol{x}) &= (4\nu-3)\mathrm{Re}\left[\sum_{h\in H_i}\sum_{k=0}^{p} f(\boldsymbol{x}-\boldsymbol{C}_h;k)\,A_1(G_h,k)\right] \\ &\quad +\mathrm{Re}\left[\sum_{h\in H_i}\sum_{k=0}^{p} f^{(\mathrm{r})}(\boldsymbol{x}-\boldsymbol{C}_h;k+1)\,a_1(G_h,k)\right] \\ &\quad -\mathrm{Re}\left[\sum_{h\in H_i}\sum_{k=0}^{p} f(\boldsymbol{x}-\boldsymbol{C}_h;k+1)\,a^{(\mathrm{r})}_1(G_h,k)\right], \end{aligned} \tag{17}$$

etc. $J_{\alpha\beta}(\boldsymbol{x})$ can also be calculated in a similar way.

Next, consider the calculation of the stress $\sigma_{\alpha\beta}(\boldsymbol{x})$ at an interior point $\boldsymbol{x}$. From (2), in order to calculate $\sigma_{\alpha\beta}(\boldsymbol{x})$, it is sufficient to calculate

$$\frac{\partial u_\alpha}{\partial x_\beta}(\boldsymbol{x}),$$

which, in turn, can be derived from

$$\frac{\partial I_{\alpha\gamma}(\boldsymbol{x})}{\partial x_\beta}.$$

The contribution from $\Gamma^{(\mathrm{f})}(\boldsymbol{x})$ of the above term can be obtained by differentiating (17) by $\boldsymbol{x}$, for instance,

$$\begin{aligned}
& 8\pi\mu(1-\nu)\frac{\partial I_{11}^{(\mathrm{f})}(\boldsymbol{x})}{\partial x_1} \\
= \; & (4\nu-3)\mathrm{Re}\left[\sum_{h\in H_i}\sum_{k=0}^{p}(-k)f(\boldsymbol{x}-\boldsymbol{C}_h;k+1)\,A_1(G_h,k)\right] \\
& +\mathrm{Re}\left[\sum_{h\in H_i}\sum_{k=0}^{p}\left\{f(\boldsymbol{x}-\boldsymbol{C}_h;k+1)-(k+1)f^{(\mathrm{r})}(\boldsymbol{x}-\boldsymbol{C}_h;k+2)\right\}a_1(G_h,k)\right] \\
& +\mathrm{Re}\left[\sum_{h\in H_i}\sum_{k=0}^{p}(k+1)f(\boldsymbol{x}-\boldsymbol{C}_h;k+2)\,a_1^{(\mathrm{r})}(G_h,k)\right],
\end{aligned}$$

where it is important to note that $A_\beta(G_h,k)$, $a_\beta(G_h,k)$ and $a_\beta^{(\mathrm{r})}(G_h,k)$ are independent of the variable $\boldsymbol{x}$.

3.6. Analytical formulae for the coefficients. The coefficients $a_\beta(G_h,k)$, $a_\beta^{(\mathrm{r})}(G_h,k)$, $a_\beta^{(\mathrm{i})}(G_h,k)$, $b_\beta(G_h,k)$, $b_\beta^{(\mathrm{r})}(G_h,k)$, $b_\beta^{(\mathrm{i})}(G_h,k)$, $b_\beta^{(\mathrm{rr})}(G_h,k)$, $b_\beta^{(\mathrm{ii})}(G_h,k)$ defined in section 3.2 and 3.4 can be calculated analytically, if constant straight line elements are used for the discretization. Of course, this is more accurate and efficient compared to numerical integration.

Let $\boldsymbol{X}_j$ and $\boldsymbol{X}_j'$ be the end points of the straight line element Γ_j, and let $\boldsymbol{Y}_j \equiv \boldsymbol{X}_j - \boldsymbol{C}_h$, $\boldsymbol{Z}_j \equiv \boldsymbol{X}_j' - \boldsymbol{C}_h$. Then, we have

$$\int_{\Gamma_j}(\boldsymbol{y}-\boldsymbol{C}_h)^k\,\mathrm{d}\Gamma(\boldsymbol{y}) \;=\; \frac{|\boldsymbol{Z}_j-\boldsymbol{Y}_j|}{\boldsymbol{Z}_j-\boldsymbol{Y}_j}\frac{1}{k+1}\left(\boldsymbol{Z}_j^{k+1}-\boldsymbol{Y}_j^{k+1}\right).$$

If we define

$$a_\beta(\Gamma_j,k) \equiv \int_{\Gamma_j}(\boldsymbol{y}-\boldsymbol{C}_h)^k q_{j\beta}\mathrm{d}\Gamma(\boldsymbol{y}), \tag{18}$$

we have

$$a_\beta(\Gamma_j,k) \;=\; \frac{|\boldsymbol{Z}_j-\boldsymbol{Y}_j|}{\boldsymbol{Z}_j-\boldsymbol{Y}_j}\frac{1}{k+1}\left(\boldsymbol{Z}_j^{k+1}-\boldsymbol{Y}_j^{k+1}\right)q_{j\beta}\,, \tag{19}$$

$$a_\beta(G_h,k) \;=\; \sum_{\Gamma_j\subset G_h} a_\beta(\Gamma_j,k). \tag{20}$$

Similarly, if we define

$$a_\beta^{(\mathrm{r})}(\Gamma_j,k) \equiv \int_{\Gamma_j} (\boldsymbol{y}-\boldsymbol{C}_h)^k \mathrm{Re}(\boldsymbol{y}-\boldsymbol{C}_h) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}), \tag{21}$$

$$a_\beta^{(\mathrm{i})}(\Gamma_j,k) \equiv \int_{\Gamma_j} (\boldsymbol{y}-\boldsymbol{C}_h)^k \mathrm{Im}(\boldsymbol{y}-\boldsymbol{C}_h) q_{j\beta} \mathrm{d}\Gamma(\boldsymbol{y}), \tag{22}$$

we obtain

$$\begin{aligned} a_\beta^{(\mathrm{r})}(\Gamma_j,k) &= \frac{\mathrm{Re}(\boldsymbol{Z}_j-\boldsymbol{Y}_j)}{\boldsymbol{Z}_j-\boldsymbol{Y}_j} a_\beta(\Gamma_j,k+1) \\ &\quad + \frac{(\boldsymbol{Z}_j-\boldsymbol{Y}_j)\mathrm{Re}(\boldsymbol{Z}_j+\boldsymbol{Y}_j) - \mathrm{Re}(\boldsymbol{Z}_j-\boldsymbol{Y}_j)(\boldsymbol{Z}_j+\boldsymbol{Y}_j)}{2(\boldsymbol{Z}_j-\boldsymbol{Y}_j)} a_\beta(\Gamma_j,k), \qquad (23) \\ a_\beta^{(\mathrm{r})}(G_h,k) &= \sum_{\Gamma_j \subset G_h} a_\beta^{(\mathrm{r})}(\Gamma_j,k). \end{aligned}$$

$a_\beta^{(\mathrm{i})}(\Gamma_j,k)$ is calculated similarly by replacing Re by Im. Further, we have

$$b_\beta(G_h,k) = \sum_{\Gamma_j \subset G_h} b_\beta(\Gamma_j,k), \qquad b_\beta(\Gamma_j,k) = \boldsymbol{n} a_\beta(\Gamma_j,k).$$

Formulae for the other coefficients can be derived similarly.

4. Theoretical estimation of the error due to the multipole expansion. Let $\hat{a}_\beta(\Gamma_j,k)$, $\hat{A}_\beta(\Gamma_j,k)$, $\hat{a}_\beta^{(\mathrm{r})}(\Gamma_j,k)$, $\hat{a}_\beta^{(\mathrm{i})}(\Gamma_j,k)$ be the values of $a_\beta(\Gamma_j,k)$, $A_\beta(\Gamma_j,k)$, $a_\beta^{(\mathrm{r})}(\Gamma_j,k)$, $a_\beta^{(\mathrm{i})}(\Gamma_j,k)$, respectively, for the case $q_{j\beta} \equiv 1$. Then, the entry $g_{i1,j1}$ of the matrix G satisfies

$$\begin{aligned} 8\pi\mu(1-\nu) g_{i1,j1} &= (4\nu-3)\mathrm{Re}\left\{ \sum_{k\geq 0} f(\boldsymbol{x}_i-\boldsymbol{C}_h;k)\, \hat{A}_1(\Gamma_j,k) \right\} \\ &+\mathrm{Re}\left\{ \sum_{k\geq 0} f^{(\mathrm{r})}(\boldsymbol{x}_i-\boldsymbol{C}_h;k+1)\, \hat{a}_1(\Gamma_j,k) \right\} - \mathrm{Re}\left\{ \sum_{k\geq 0} f(\boldsymbol{x}_i-\boldsymbol{C}_h;k+1)\, \hat{a}_1^{(\mathrm{r})}(\Gamma_j,k) \right\}. \end{aligned}$$

The multipole method truncates the above infinite series at the pth term. Then the error $\Delta G \equiv (\Delta g_{i\alpha,j\beta})$ in approximating G by the multipole method satisfies the equation

$$\begin{aligned} 8\pi\mu(1-\nu)\Delta g_{i1,j1} &= (4\nu-3)\mathrm{Re}\left\{ \sum_{k>p} f(\boldsymbol{x}_i-\boldsymbol{C}_h;k)\, \hat{A}_1(\Gamma_j,k) \right\} \\ &\quad +\mathrm{Re}\left\{ \sum_{k>p} f^{(\mathrm{r})}(\boldsymbol{x}_i-\boldsymbol{C}_h;k+1)\, \hat{a}_1(\Gamma_j,k) \right\} \\ &\quad -\mathrm{Re}\left\{ \sum_{k>p} f(\boldsymbol{x}_i-\boldsymbol{C}_h;k+1)\, \hat{a}_1^{(\mathrm{r})}(\Gamma_j,k) \right\}. \end{aligned} \tag{24}$$

In the following, we derive an upper bound for $\Delta g_{i1,j1}$. By physical considerations, we have $0 < \nu < 1/2$, so that

$$|4\nu-3| < 3. \tag{25}$$

The inequalities

$$\frac{|\boldsymbol{Z}_j|}{|\boldsymbol{x}_i - \boldsymbol{C}_h|} < \frac{1}{2}, \quad \frac{|\boldsymbol{Y}_j|}{|\boldsymbol{x}_i - \boldsymbol{C}_h|} < \frac{1}{2}$$

and (19) give

$$\begin{aligned} |f(\boldsymbol{x}_i - \boldsymbol{C}_h; k)\, \hat{a}_\beta(\Gamma_j, k)| &= \frac{|\boldsymbol{Z}_j - \boldsymbol{Y}_j|}{k+1} \left| \frac{\boldsymbol{Z}_j^{k+1} - \boldsymbol{Y}_j^{k+1}}{(\boldsymbol{Z}_j - \boldsymbol{Y}_j)(\boldsymbol{x}_i - \boldsymbol{C}_h)^k} \right| \\ &< \frac{\text{radius}(\Gamma_j)}{2^{k-1}}\,. \end{aligned} \tag{26}$$

Since $\hat{A}_\beta(\Gamma_j, k) = -\hat{a}_\beta(\Gamma_j, k)/k$, we have

$$|f(\boldsymbol{x}_i - \boldsymbol{C}_h; k)\, \hat{A}_\beta(\Gamma_j, k)| < \frac{\text{radius}(\Gamma_j)}{k 2^{k-1}}, \qquad k \neq 0. \tag{27}$$

Thus, we obtain the inequality

$$|f^{(\mathrm{r})}(\boldsymbol{x}_i - \boldsymbol{C}_h; k+1)\, \hat{a}_\beta(\Gamma_j, k)| \leq |f(\boldsymbol{x}_i - \boldsymbol{C}_h; k)\, \hat{a}_\beta(\Gamma_j, k)| < \frac{\text{radius}(\Gamma_j)}{2^{k-1}}\,. \tag{28}$$

Moreover, from (23) we obtain

$$\begin{aligned} & |f(\boldsymbol{x}_i - \boldsymbol{C}_h; k+1)\, \hat{a}^{(\mathrm{r})}_\beta(\Gamma_j, k)| \\ \leq\ & |f(\boldsymbol{x}_i - \boldsymbol{C}_h; k+1) a_\beta(G_h, k+1)| \\ & + \frac{|\boldsymbol{Z}_j - \boldsymbol{Y}_j|\, |\boldsymbol{Z}_j + \boldsymbol{Y}_j|}{2|(\boldsymbol{Z}_j - \boldsymbol{Y}_j)(\boldsymbol{x}_i - \boldsymbol{C}_h)|} |f(\boldsymbol{x}_i - \boldsymbol{C}_h; k) \hat{a}_\beta(G_h, k)|. \end{aligned}$$

Thus, (26) gives

$$\begin{aligned} |f(\boldsymbol{x}_i - \boldsymbol{C}_h; k+1)\, \hat{a}^{(\mathrm{r})}_\beta(\Gamma_j, k)| &< \text{radius}(\Gamma_j)/2^k + \text{radius}(\Gamma_j)/2^k \\ &= \text{radius}(\Gamma_j)/2^{k-1}. \end{aligned} \tag{29}$$

Hence, (24), (25), (26), (27), (28) and (29) give

$$\begin{aligned} |8\pi\mu(1-\nu)\Delta g_{i1,j1}| &< \sum_{k>p} \left(\frac{3}{k} + 1 + 1 \right) \text{radius}(\Gamma_j)/2^{k-1} \\ &< \left(\frac{3}{p+1} + 2 \right) \text{radius}(\Gamma_j)/2^{p-1}. \end{aligned}$$

Noting that $0 < \nu < 1/2$, we obtain

$$|\Delta g_{i1,j1}| < \frac{\text{radius}(\Gamma_j)}{\pi\mu} \left(\frac{3}{p+1} + 2 \right) \frac{1}{2^{p+1}}.$$

Similar arguments lead to

$$|\Delta g_{i\alpha,j\beta}| < \frac{\text{radius}(\Gamma_j)}{\pi\mu} \left(\frac{3\delta_{\alpha\beta}}{p+1} + 2 \right) \frac{1}{2^{p+1}}$$

for general α and β. Hence, we arrive at

$$
\begin{aligned}
|\Delta G|_\infty &= \max_{i,\alpha} \sum_{j,\beta} |\Delta g_{i\alpha,j\beta}| < \max_{i,\alpha} \sum_{j,\beta} \frac{\mathrm{radius}(\Gamma_j)}{\pi\mu} \left(\frac{3\delta_{\alpha\beta}}{p+1} + 2 \right) \frac{1}{2^{p+1}} \\
&\leq \frac{L}{\pi\mu} \left(\frac{3}{p+1} + 4 \right) \frac{1}{2^{p+1}}, \qquad (30)
\end{aligned}
$$

where L is the perimeter of the boundary Γ.

Similarly, an upper bound for $\Delta H \equiv (\Delta h_{i\alpha,j\beta})$, which is the error in the approximation of H is given by

$$
|h_{i\alpha,j\beta}| < \frac{(3p+8)\mathrm{radius}(\Gamma_j)}{2^{p+1}\pi|\boldsymbol{x}_i - \boldsymbol{C}_h|},
$$

so that

$$
|\Delta H|_\infty = \max_{i,\alpha} \sum_{j,\beta} |h_{i\alpha,j\beta}| \quad < \quad \frac{3p+8}{2^p\pi} \max_i \sum_{h\in H_i} \sum_{\Gamma_j \subset G_h} \frac{\mathrm{radius}(\Gamma_j)}{|\boldsymbol{x}_i - \boldsymbol{C}_h|}.
$$

If the number of elements N is sufficiently large, most clusters G_h, except for a constant number of large clusters, can be regarded as straight line segments. For these clusters G_h, we may assume that

$$
\sum_{\Gamma_j \subset G_h} \mathrm{radius}(\Gamma_j) \approx \mathrm{radius}(G_h).
$$

As discussed in section 3.3, the number of clusters which each cluster list H_i contains is $\mathrm{O}(\log N)$, and the definition of the cluster guarantees the inequality

$$
\frac{\mathrm{radius}(\Gamma_j)}{|\boldsymbol{x}_i - \boldsymbol{C}_h|} < \frac{1}{2}.
$$

Hence,

$$
\sum_{h\in H_i} \sum_{\Gamma_j \subset G_h} \frac{\mathrm{radius}(\Gamma_j)}{|\boldsymbol{x}_i - \boldsymbol{C}_h|} = \mathrm{O}(\log N) \frac{\mathrm{radius}(G_h)}{|\boldsymbol{x}_i - \boldsymbol{C}_h|} = \mathrm{O}(\log N)
$$

holds.

Based on the discussions above, we will consider how to determine the number of terms p for the multipole expansion. Let us restrict the problem to the Dirichlet problem. Consider the inequality [7]

$$
\frac{|\Delta \boldsymbol{U}|_\infty}{|\boldsymbol{U}|_\infty} \leq \frac{|G|_\infty |G^{-1}|_\infty}{1 - |\Delta G|_\infty |G^{-1}|_\infty} \left(\frac{|\Delta G|_\infty}{|G|_\infty} + R \right), \qquad (31)
$$

where $|\cdot|_\infty$ stands for the max norm of a vector and the max norm of a matrix derived from the vector norm. $\Delta \boldsymbol{U}$ stands for the error of the solution obtained by the multipole method, compared to the true solution of (6). R can be split into two components, i.e.,

$$
R = R^{(\mathrm{I})} + R^{(\mathrm{C})}, \qquad (32)
$$

where $R^{(\mathrm{I})}$ is the relative residual due to the iterative method, and $R^{(\mathrm{C})}$ is the relative residual due to the approximate calculation of $H\boldsymbol{U}$ using the multipole method. (31) and (32) suggest that $|\Delta G|_\infty/|G|_\infty$, $R^{(\mathrm{I})}$ and $R^{(\mathrm{C})}$ should be of the same order for efficiency. Hence, we will consider how to choose p suitably when $R^{(\mathrm{I})}$ is given. Since

$$R^{(\mathrm{C})} \leq \frac{|\Delta H\boldsymbol{U}|_\infty}{|H\boldsymbol{U}|_\infty} \leq \frac{|\Delta H|_\infty|\boldsymbol{U}|_\infty}{|H\boldsymbol{U}|_\infty} < \frac{(3p+8)|\boldsymbol{U}|_\infty}{2^p\pi|H\boldsymbol{U}|_\infty} \max_i \sum_{h\in H_i} \sum_{\Gamma_j\subset G_h} \frac{\mathrm{radius}(\Gamma_j)}{|\boldsymbol{x}_i-\boldsymbol{C}_h|},$$

p should be chosen so that

$$R^{(\mathrm{I})} \approx \frac{(3p+8)|\boldsymbol{U}|_\infty}{2^p\pi|H\boldsymbol{U}|_\infty} \max_i \sum_{h\in H_i} \sum_{\Gamma_j\subset G_h} \frac{\mathrm{radius}(\Gamma_j)}{|\boldsymbol{x}_i-\boldsymbol{C}_h|}, \tag{33}$$

when calculating $H\boldsymbol{U}$. On the other hand, since

$$\frac{|\Delta G|_\infty}{|G|_\infty} \leq \frac{|\Delta G|_\infty|\boldsymbol{P}|_\infty}{|G\boldsymbol{P}|_\infty} < \frac{L|\boldsymbol{P}|_\infty}{\pi\mu|G\boldsymbol{P}|_\infty}\left(\frac{3}{p+1}+4\right)\frac{1}{2^{p+1}},$$

p should be chosen so that

$$R^{(\mathrm{I})} \approx \frac{L|\boldsymbol{P}|_\infty}{\pi\mu|G\boldsymbol{P}|_\infty}\left(\frac{3}{p+1}+4\right)\frac{1}{2^{p+1}} \approx \frac{L|\boldsymbol{P}|_\infty}{\pi\mu|H\boldsymbol{U}|_\infty}\frac{1}{2^{p-1}} \tag{34}$$

when calculating $G\boldsymbol{q}$, where $\boldsymbol{q}$ is the temporary vector. We can argue in a similar way for the Neumann and mixed problems, and conclude that (33) should hold when calculating $H\boldsymbol{q}$, and (34) should hold when calculating $G\boldsymbol{q}$. The only difference is that

$$\max_i \sum_{h\in H_i}$$

in the right hand side of (33) should be replaced by

$$\max_{i,\alpha} \sum_{h\in H_{i\alpha}},$$

when considering a mixed problem. The above theoretical error estimates, though pessimistic, were reflected in the numerical experiments.

5. Numerical experiment results. In this section we will give numerical experiment results for two dimensional elastostatic problems, comparing the proposed multipole method with previous methods. The computations were done on the workstation DEC 3000 model AXP (150 MHz) with 64 Mbyte main memory and 200 Mbyte swap area.

The term 'simple iteration' will indicate the results obtained by the usual iterative methods such as the Bi-CGSTAB [6] or the GCR (GMRES) [5] without using the multipole method.

The convergence threshold for the relative residual of the iterative methods (with or without multipole method) was set to $R^{(\mathrm{I})} = 10^{-5}$. The multipole expansions were truncated at $p = 25$. The discretization of the boundary was done using constant straight line elements.

5.1. Dirichlet problem. First, we will give results for a Dirichlet problem. The domain and the boundary conditions are given by

$$\begin{aligned} \Omega &\equiv \left\{ \boldsymbol{x} | x_1^2 + x_2^2 \leq 9 \right\}, \\ u(\boldsymbol{x}) &= -10^{-6} \times x_2 \qquad \boldsymbol{x} \in \Gamma. \end{aligned}$$

The elastic constants were set to $\nu = 0.1$ and $\mu = 9.45 \times 10^4$, where the length is measured in meters and the Lamé constants are in MPa.

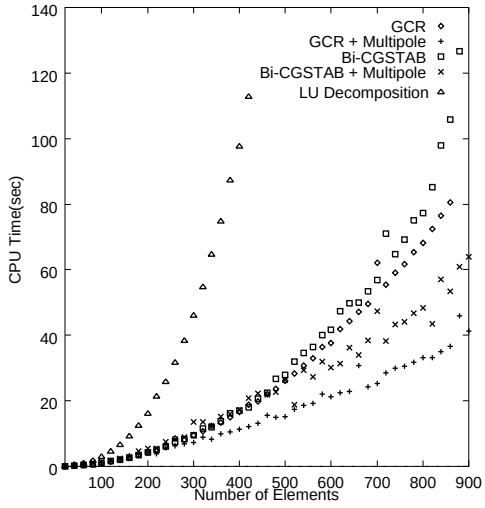

FIG. 4. *Comparison of the CPU time for the multipole method, the simple iteration and LU decomposition.*

Fig. 4 compares the CPU time for the multipole method, simple iteration and LU decomposition, as the number of boundary elements increases. The LU decomposition consumed the most CPU time, regardless of the number of elements N. For N larger than 500, the second slowest algorithm was the Bi-CGSTAB by simple iteration, the third was the GCR by simple iteration, the fourth was the Bi-CGSTAB by the multipole method, and GCR by the multipole method was the fastest.

5.2. Neumann problem for a domain with many holes. Next, a Neumann problem defined for a domain Ω with many circular holes is considered, where

$$\begin{aligned} \Omega &\equiv \{\boldsymbol{x} | -2 \leq x_1 \leq 2,\ -1 \leq x_2 \leq 1\} - \sum_{m=1}^{8} \sum_{n=1}^{4} C_{lm}, \\ C_{lm} &\equiv \left\{ \boldsymbol{x} \left| \{9x_1 - (4m - 18)\}^2 + \{9x_2 - (2n - 5)\}^2 < 0.81 \right. \right\} \end{aligned}$$

as shown in Fig. 5. The boundary condition is given by

$$\boldsymbol{p}(\boldsymbol{x}) = \begin{cases} (1,\ 0)^{\mathrm{T}} & x_1 = \pm 2, \\ \mathbf{0} & \text{otherwise}. \end{cases}$$

The elastic constants are the same as in the previous example.

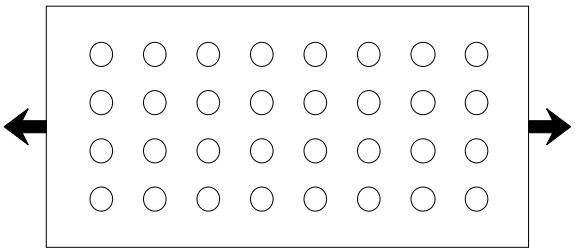

FIG. 5. *Domain with many holes.*

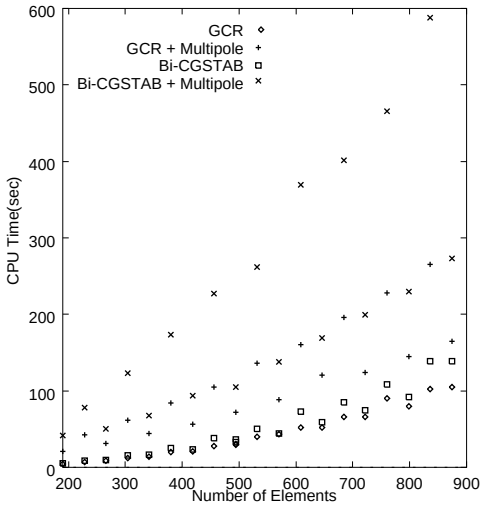

FIG. 6. *Comparison of CPU time between multipole method and simple iteration* ($R^{(\mathrm{I})} = 10^{-5}$, $p = 25$).

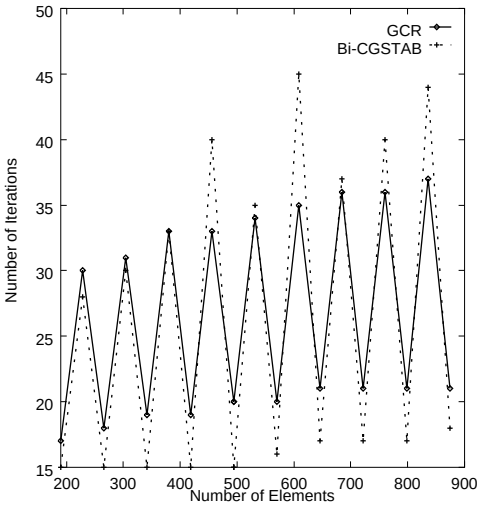

FIG. 7. *Number of iterations* (*simple iteration,* $R^{(\mathrm{I})} = 10^{-5}$).

This is a model problem of a porous media, and suggests that the number of boundary elements N can become quite large, even for two dimensional problems, when the geometry of the boundary Γ is complex. As shown in Fig. 6, simple iteration is faster than the multipole method when N is smaller than 900. Fig. 7 shows the number of iterations with respect to the number of elements N. Only the results for the simple iterations is shown, since the difference between the simple iteration and the multipole method is very small. This problem is very ill-conditioned and requires many iterations compared to the previous Dirichlet problem. Also the convergence behavior is oscillatory with the increase of N, depending on the discretization of the holes. In the multipole method, the multipole expansions have to be performed for each iteration, whereas for the simple iteration, the G and H matrices are computed only once and can be shared among the iterations. Hence, the more iterations required to solve the linear systems, the lighter the weight of the CPU time for the construction of G and H matrices becomes. This is the reason why the simple iteration is relatively fast for this problem. However, when the number of elements exceeds 800, the data for the simple iterations can no longer be saved on the main memory, and frequent access to the disk makes the elapsed time far longer compared to the CPU time. Fig. 8 compares the elapsed time for the simple iteration and the multipole method for

the same problem.

In order to make the multipole method more competitive and robust, we need an efficient method to reduce the number of iterations. Preliminary attempts using diagonal scaling or ILU decomposition did not improve the situation substantially.

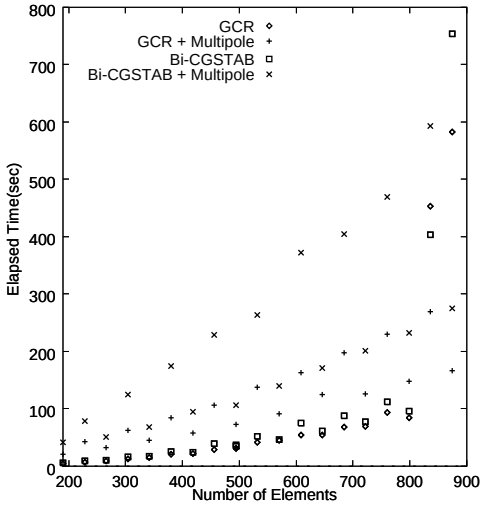

FIG. 8. *Comparison of the elapsed time between the multipole method and the simple iteration* ($R^{(\mathrm{I})} = 10^{-5}$, $p = 25$).

6. Conclusions. We proposed a multipole method for economizing the solution of the two dimensional elastostatic problem using the boundary element method.

Preliminary experiments showed that the method reduces the CPU time and memory remarkably. However, for ill-conditioned problems which require many iterations, the method becomes less competitive in CPU time, although it is still competitive in terms of elapsed time, which is determined by the required memory.

The search for an effective method for reducing the number of iterations, and a more efficient implementation of the multipole method are left for future work.

Acknowledgements. The authors would like to thank Professor K. Sugihara, Dr. M. Sugihara and Mr. T. Imai for valuable comments.

REFERENCES

[1] A. BREBBIA AND J. DOMINGUEZ, *Boundary Elements An Introductory Course*, Computaional Mechanics Publications with McGraw-Hill Book Company, New York, 1992.

[2] A. GREENBAUM, L. GREENGARD, AND G. B. MCFADDEN, *Laplace's equation and Dirichlet-Neumann map in multiply connected domains*, J. Comput. Phys., 105 (1993), pp. 267–278.

[3] L. GREENGARD, *The Rapid Evaluation of Potential Fields in Particle Systems*, MIT Press, Cambridge, 1988.

[4] V. ROKHLIN, *Rapid solution of integral equations of classical potential theory*, J. Comput. Phys., 60 (1985), pp. 187–207.

[5] Y. SAAD AND M. H. SCHULTZ, *GMRES: A generalized minimum residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856–869.

[6] H. A. VAN DER VORST, *Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., 13 (1992), pp. 631–644.

[7] D. S. WATKINS, *Fundamentals of Matrix Computations*, Wiley, 1991.