Performance Improvement of Extended Boundary Node Method for Solving Elliptic Boundary-Value Problems*)

Ayumu SAITOH, Teruou TAKAYAMA[1)], Atsushi KAMITANI and Hiroaki NAKAMURA[2)]

Graduate School of Science and Engineering, Yamagata University, Yonezawa 992-8510, Japan

[1)]*Department of Informatics, Yamagata University, Yonezawa 992-8510, Japan*

[2)]*National Institute for Fusion Science, Toki 509-5292, Japan*

(Received 30 November 2015 / Accepted 14 March 2016)

The extended boundary-node method (X-BNM) with the hierarchical-matrix ($\mathscr{H}$-matrix) method has been developed and its performance has been investigated numerically. The results of computations show that the solver speed of the X-BNM with the $\mathscr{H}$-matrix method is much faster than that of the standard X-BNM for the case where the number of boundary nodes exceeds a certain limit. Furthermore, the accuracy of the X-BNM with the $\mathscr{H}$-matrix method is almost equal to that of the standard X-BNM. From these results, it is found that the $\mathscr{H}$-matrix method is useful as the acceleration technique of the X-BNM.

Keywords: elliptic boundary-value problems, boundary integral equations, meshless method, Krylov subspace methods, $\mathscr{H}$-matrix, numerical analysis

DOI: 10.1585/pfr.11.2401062

1. Introduction

Recently, the boundary node method (BNM) [1], which is one of boundary-type meshless methods, has been proposed. As the feature of the BNM, a boundary does not need to be divided into a set of elements before executing the BNM code. In addition, a smooth numerical solution is obtained because the shape function is determined by using the moving least-squares approximation. However, it is the inherent demerit of the BNM that integration cells must be used for calculating matrix elements.

In order to resolve the above demerit, the BNM has been reformulated without using integration cells. Throughout the present study, the method is called the extended BNM (X-BNM) [2, 3]. The results of computations have shown that the accuracy of the X-BNM is much higher than that of the dual reciprocity boundary element method [4]. In addition, we have also shown that the number of unknowns can be reduced to half by using the RPIM shape function which has the Kronecker delta function property [5].

On the other hands, the hierarchical-matrix ($\mathscr{H}$-matrix) method [6–8] has been proposed for accelerating the calculation speed of the matrix-vector product. By using the method, it is possible to reduce the computational cost from $O(N^2)$ to $O(N) \sim O(N \log N)$, where N denotes the number of boundary nodes. If the $\mathscr{H}$-matrix method were applied to the X-BNM, the speed of the X-BNM could be further accelerated.

The purpose of the present study is to apply the $\mathscr{H}$-matrix method to the X-BNM and to investigate the performance of the proposed method numerically. In this study, the boundary-value problem of the Grad-Shafranov (G-S) equation is used as the elliptic boundary-value problems.

2. Extended Boundary-Node Method

2.1 Boundary integral equation and discretization

Let us first derive a boundary integral equation from the G-S equation. As an example, we consider a boundary-value problem of the G-S equation on the domain Ω bounded by a simple closed curve $\partial\Omega$ in the cylindrical coordinate (r, z):

$$-\hat{L}\psi = 0 \qquad \text{in } \Omega, \tag{1}$$

$$\psi = \bar{\psi} \qquad \text{on } \Gamma_{\mathrm{D}}, \tag{2}$$

$$\frac{\partial \psi}{\partial n} \equiv q = \bar{q} \qquad \text{on } \Gamma_{\mathrm{N}}, \tag{3}$$

where Γ_{D} and Γ_{N} are parts of $\partial\Omega$ such that $\Gamma_{\mathrm{D}} \cup \Gamma_{\mathrm{N}} = \partial\Omega$ and $\Gamma_{\mathrm{D}} \cap \Gamma_{\mathrm{N}} = \phi$. Moreover, $\bar{\psi}$ and $\bar{q}$ are given functions on Γ_{D} and Γ_{N}, respectively. Furthermore, $\boldsymbol{n}$ indicates an outward unit normal on $\partial\Omega$. In addition, $\hat{L}$ denotes the G-S operator and its explicit form is defined by

$$\hat{L} \equiv \frac{\partial^2}{\partial z^2} + r\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial}{\partial r}\right).$$

From the straightforward calculation, we can show that (1) is equivalent to the following boundary integral equation:

$$\oint_{\partial\Omega} \frac{1}{r}\left(w^* q(\boldsymbol{x}(s)) - \frac{\partial w^*}{\partial n}\left[\psi(\boldsymbol{x}(s)) - \psi(\boldsymbol{y})\right]\right) \mathrm{d}s = 0, \tag{4}$$

author's e-mail: saitoh@yz.yamagata-u.ac.jp

*) This article is based on the presentation at the 25th International Toki Conference (ITC25).

where $w^*(\boldsymbol{x}(s), \boldsymbol{y})$ and $\partial w^*(\boldsymbol{x}(s), \boldsymbol{y})/\partial n$ denote the fundamental solution of $-\hat{L}\psi = r\delta(\boldsymbol{x}(s) - \boldsymbol{y})$ and its normal derivative, respectively. Moreover, s indicates an arclength along $\partial\Omega$.

Next, we discretize the boundary integral equation (4) and the associated boundary conditions (2) and (3). To this end, N boundary nodes, $\boldsymbol{x}(s_1), \boldsymbol{x}(s_2), \cdots, \boldsymbol{x}(s_N)$, are placed on $\partial\Omega$ and, subsequently, the ith RPIM shape function $\phi_i(s)$ [5, 9] is assigned to the ith boundary nodes. Furthermore, ψ and q are assumed as

$$\psi(\boldsymbol{x}(s)) = \sum_{i=1}^{N} \phi_i(s)\,\psi_i,$$
$$q(\boldsymbol{x}(s)) = \sum_{i=1}^{N} \phi_i(s)\,q_i,$$

where ψ_i and q_i are the solution on ith boundary node and its normal derivative, respectively.

Under the aforementioned assumptions, (4) and the associated boundary conditions are discretized as the following linear system:

$$G\boldsymbol{q} - H\boldsymbol{u} = \boldsymbol{0}. \tag{5}$$

Here, $G, H, \boldsymbol{q}$ and $\boldsymbol{u}$ are defined by

$$G = \sum_{i=1}^{N}\sum_{j=1}^{N} g_{ij}\,\boldsymbol{e}_i\,\boldsymbol{e}_j^T,$$
$$H = \sum_{i=1}^{N}\sum_{j=1}^{N} h_{ij}\,\boldsymbol{e}_i\,\boldsymbol{e}_j^T,$$
$$\boldsymbol{q} = \sum_{i=1}^{N} q_i\,\boldsymbol{e}_i,$$
$$\boldsymbol{u} = \sum_{i=1}^{N} \psi_i\,\boldsymbol{e}_i,$$

where the influence coefficients, h_{ij} and g_{ij}, are defined by

$$h_{ij} = \oint_{\partial\Omega} \frac{1}{r}\frac{\partial w^*(\boldsymbol{x}(s), \boldsymbol{x}(s_i))}{\partial n}\left[\phi_j(s) - \delta_{i,j}\right]\mathrm{d}s, \tag{6}$$
$$g_{ij} = \oint_{\partial\Omega} \frac{1}{r} w^*(\boldsymbol{x}(s), \boldsymbol{x}(s_i))\,\phi_j(s)\mathrm{d}s. \tag{7}$$

Furthermore, $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_N\}$ is the orthonormal system of the N-dimensional vector space.

Reordering (5) such that all unknowns are on the left-hand side, the above linear system can be rewritten as

$$A\boldsymbol{x} = \boldsymbol{b}, \tag{8}$$

where A denotes the $N \times N$ coefficient matrix obtained by reordering (5). Moreover, $\boldsymbol{x}$ and $\boldsymbol{b}$ are the N-dimensional solution vector and the N-dimensional vector generated by using known boundary conditions, respectively. It is (8) that the discretization form of the boundary-value problem of the G-S equation. By solving (8), we can easily get the solution and its normal derivative on $\partial\Omega$. In the next section, we explain how to calculate h_{ij} and g_{ij} without using the integration cell.

2.2 Calculation of influence coefficients

In the standard BNM, we must divide the boundary into a set of integration cells because of the calculation of (6) and (7). However, this means that a concept of boundary elements is partly contained in spite of one of meshless methods.

In the X-BNM, the influence coefficients, h_{ij} and g_{ij}, are directly calculated by use of a vector equation of $\partial\Omega$. In fact, it is determined by means of the following three steps:

1) The isosurface $f(\boldsymbol{x}) = 0$ of the boundary is determined for the curve passing through $\partial\Omega$.
2) The following ordinary differential equation:
$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}s} = \boldsymbol{R}\left(\frac{\pi}{2}\right)\cdot\frac{\nabla f}{|\nabla f|},$$
is solved numerically. Here, $\boldsymbol{R}(\theta)$ denotes a tensor representing a rotation through an angle θ.
3) The resulting P data points, $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \cdots, \boldsymbol{x}^{(P)}$, are interpolated with the cubic spline. Here, the detailed explanation for obtaining P data points is shown in [4].

Since the vector equation of $\partial\Omega$ is represented as a function of s, we can easily calculate the influence coefficients without using integration cells.

3. Fast Calculation of Matrix-Vector Product

The coefficient matrix A becomes asymmetric and dense. In addition, it has not a diagonal-dominant. Therefore, we cannot solve (8) by using stationary iterative methods. For this reason, the Krylov subspace methods have been adopted as the solver of (8). In order to further accelerate the speed of the X-BNM, the $\mathscr{H}$-matrix method is applied to the matrix-vector product in the Krylov subspace methods.

In the $\mathscr{H}$-matrix method, we first generate a hierarchical cluster structure from a location information of boundary nodes. Next, by using the structure, the coefficient matrix A is divided into a set of sub-matrices, $A^1, A^2, \cdots, A^M$, where M is the total number of clusters. When the cluster distance is near, the original sub-matrix is stored. Otherwise, it is approximated as the potential low-rank matrices by using the adaptive cross approximation.

When the ratio of the number of far clusters to the total number of clusters is large, the matrix-vector product can be computed fast. As a result, the speed of the X-BNM becomes fast. Throughout the present study, we adopt the GMRES(200) method as the solver of (8). Moreover, the convergence condition ε_{G} is fixed as $\varepsilon_{\mathrm{G}} = 10^{-12}$.

4. Numerical Results

In this section, the performance of the X-BNM with the $\mathscr{H}$-matrix method is investigated numerically. In the

following, we adopt the boundary-value problem of the G-S equation with the Dirichlet condition:

$$u = -\frac{(r^2-1)^2}{4} + r^2 z^2,$$

on $\partial\Omega$. Furthermore, $\partial\Omega$ is assumed as

$$\partial\Omega = \left\{ \boldsymbol{x} \in \boldsymbol{R}^2 \middle| f(\boldsymbol{x}) \equiv \left[x - 2 - \Delta(y/2)^2\right]^2 + (y/2)^2 - 1 = 0 \right\},$$

where Δ is the triangularity.

As the measure of the accuracy of the numerical solution, we adopt the relative error defined by

$$\epsilon = \frac{\left\| q_{\mathrm{A}} - q_{\mathrm{N}} \right\|}{\left\| q_{\mathrm{A}} \right\|},$$

where subscript notations, A and N, are analytic and numerical solutions, respectively. Moreover, $\| \ \|$ denotes an maximum norm.

In the X-BNM, the radial basis function $r_i(s)$ used in the RPIM shape function is given by

$$r_i(s) = \rho\left(|s - s_i| / R_i\right),$$
$$\rho(r) = \begin{cases} \exp(\alpha r^2) & (r \le 1), \\ 0 & (r > 1). \end{cases}$$

Here, s_i denotes the arclength from the first boundary node to the ith boundary node and R_i is defined by

$$R_i = \gamma \min\left(\left| s_{\bmod(i+1,N)} - s_i \right|, \left| s_{\bmod(i-1,N)} - s_i \right| \right),$$

where γ is a constant. Throughout the present study, the number m of terms in the basis and the parameters are fixed as follows: $m = 1$, $\alpha = 5.3$ and $\gamma = 1.1$.

Let us first investigate the accuracy of the X-BNM with the $\mathscr{H}$-matrix method. The relative errors are calculated as a function of the number N of boundary nodes and are depicted in Fig. 1. We see from this figure that the convergence rate of the X-BNM with the $\mathscr{H}$-matrix method is almost equal to that of the standard X-BNM. Furthermore, this figure indicates that the accuracy of the X-BNM with the $\mathscr{H}$-matrix method is almost equal to that of the standard X-BNM.

Next, the speed of the X-BNM with the $\mathscr{H}$-matrix method is compared with the standard X-BNM. In Fig. 2, the computation time required for the solver is measured and τ_{S} and τ_{A} are plotted as a function of N. Here, τ_{S} and τ_{A} denote the computation time of the standard X-BNM and that of the X-BNM with the $\mathscr{H}$-matrix method, respectively. This figure indicates that the speed of the X-BNM with the $\mathscr{H}$-matrix method is 0.93 - 34.8 times as much as the standard X-BNM for the case with $N \gtrsim 500$.

For investigating the above reason, the residual histories of the X-BNM with the $\mathscr{H}$-matrix method and the standard X-BNM are shown in Figs. 3 (a) and 3 (b), respectively. For the case with both methods, the residual norm

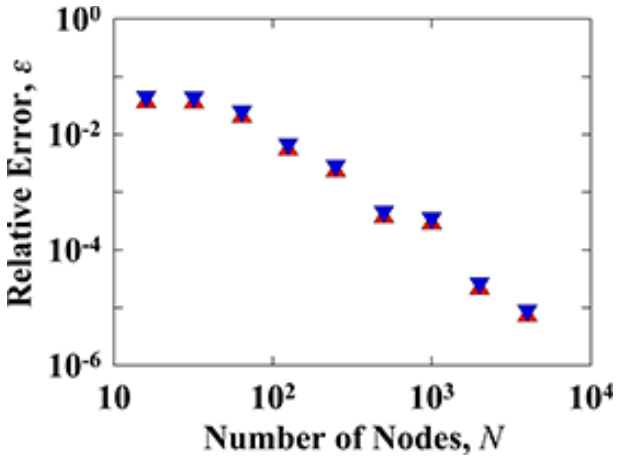


Fig. 1 Dependence of the relative error ε on the number N of boundary nodes for the case with $\Delta = 1$. Here, the blue symbol and red one denote the X-BNM with the $\mathscr{H}$-matrix method and the standard X-BNM, respectively.

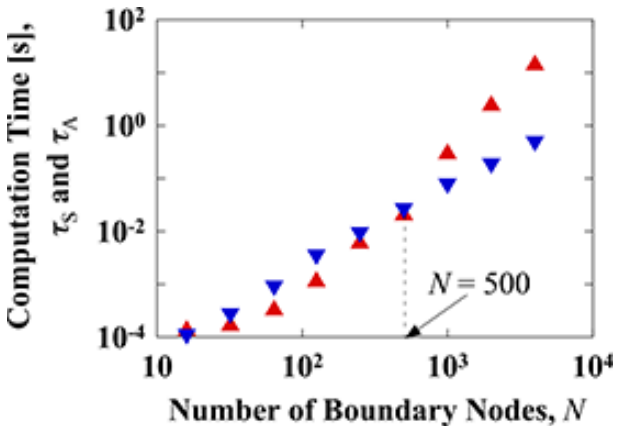
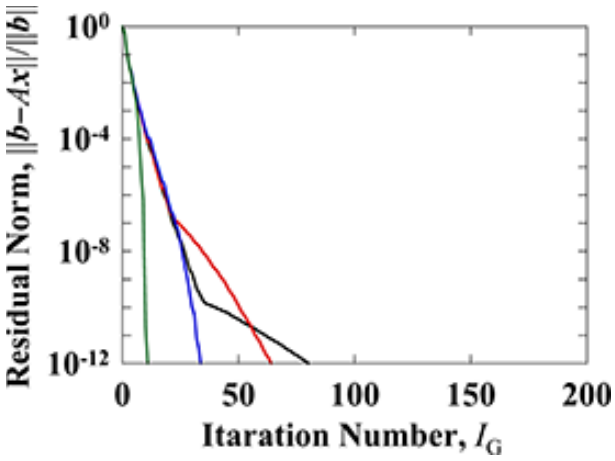


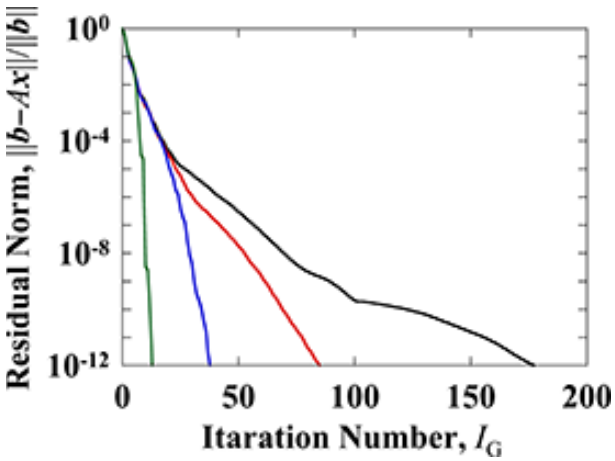
Fig. 2 Dependence of the computation time, τ_{S} and τ_{A}, on the number N of boundary nodes for the case with $\Delta = 1$. Here, the blue symbol and red one denote τ_{A} and τ_{S}, respectively.

decreases monotonously with an increase in the iteration number. This tendency does not depend on the value of N. In addition, the total iteration number of the X-BNM with the $\mathscr{H}$-matrix method is a little larger than that of the standard X-BNM. Although the total iteration number of the X-BNM with the $\mathscr{H}$-matrix method increases by comparing with the standard X-BNM, it is found that the calculation cost of the X-BNM with the $\mathscr{H}$-matrix method extremely decreases by means of the effect of the potential low-rank matrix.

Finally, we investigate the influence of the boundary shape on the accuracy of the solution. The relative errors are calculated as a function of the triangularity Δ and are depicted in Fig. 4. Both the accuracy of the X-BNM with the $\mathscr{H}$-matrix method and that of the standard X-BNM monotonously increase with an increase in Δ. Moreover, the accuracy of the X-BNM with the $\mathscr{H}$-matrix method is



(a)



(b)

Fig. 3 Residual histories of (a) the X-BNM with the $\mathscr{H}$-matrix method and (b) the standard X-BNM for the case with $\Delta = 1$. Here, black, red, blue and green lines denote $N = 4000$, $N = 500$, $N = 64$ and $N = 16$, respectively.

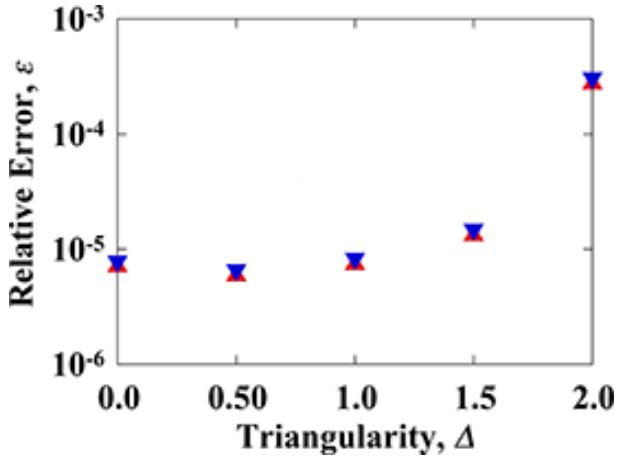


Fig. 4 Dependence of the relative error ε on the triangularity Δ ($N = 4000$). Here, the blue symbol and red one denote the X-BNM with the $\mathscr{H}$-matrix method and the standard X-BNM, respectively.

almost equal to that of the standard X-BNM regardless of the value of Δ.

From these results, we can conclude that the $\mathscr{H}$-matrix method is useful for accelerating the X-BNM.

5. Conclusion

We have applied the $\mathscr{H}$-matrix method to the X-BNM and have numerically investigated its performance by comparing with the standard X-BNM. Conclusions obtained in this paper are summarized as follows.

1. The solver speed of the X-BNM with the $\mathscr{H}$-matrix method is much faster than that of the standard X-BNM for $N \gtrsim 500$.
2. The accuracy of the X-BNM with the $\mathscr{H}$-matrix method is almost equal to that of the standard X-BNM. Even if the boundary shape is concave, this tendency does not change.

Acknowledgment

This work was supported by Japan Society for the Promotion of Science under Grant-in-Aid for Young Scientists (B) (No. 25870630) and Scientific Research (C) (No. 26520204). A part of this work was also carried out with the support and under the auspices of the NIFS Collaboration Research program (NIFS15KNTS041, NIFS15KNXN297).

[1] Y.X. Mukherjee and S. Mukherjee, Int. J. Numer. Meth. Eng. **40**, 797 (1997).
[2] A. Saitoh, S. Nakata, S. Tanaka and A. Kamitani, Information **12**, 973 (2009) [in Japanese].
[3] A. Saitoh, N. Matsui, T. Itoh and A. Kamitani, IEEE Trans. Magn. **47**, 1222 (2011).
[4] A. Saitoh, K. Miyashita, T. Itoh, A. Kamitani, T. Isokawa, N. Kamiura and N. Matsui, IEEE Trans. Magn. **49**, 1601 (2013).
[5] A. Saitoh, T. Itoh, N. Matsui and A. Kamitani, IEEE Trans. Magn. **51**, 7204704 (2015).
[6] S. Kurz, O. Rain and S. Rjasanow, IEEE Trans. Magn. **38**, 421 (2002).
[7] Y. Takahashi, C. Matsumoto and S. Wakao, IEEE Trans. Magn. **43**, 1277 (2007).
[8] V. Le-Van, B. Bannwarth, A. Carpentier, O. Chadebec, J.M. Guichon and G. Meunier, IEEE Trans. Magn. **50**, 7010904 (2014).
[9] J.G. Wang and G.R. Liu, Int. J. Numer. Meth. Eng. **54**, 1623 (2002).