Structural Eng./Earthquake Eng., JSCE, Vol.22, No. 2, 219s-231s, 2005 October
(Translated from a paper which originally appeared on Journal of Applied Mechanics,
JSCE, Vol.6, pp.301-310, 2003.8)

MATRIX-COMPRESSION PROPERTY OF BEYLKIN-TYPE TRUNCATION SCHEME FOR WAVELET BEM

Kazuhiro KORO[1] and Kazuhisa ABE[2]

[1]Member of JSCE, Research Associate, Graduate School of Science and Technology, Niigata University
(8050 Igarashi 2-Nocho, Niigata, 950-2181, JAPAN)
E-mail: kouro@eng.niigata-u.ac.jp
[2]Member of JSCE, Associate Professor, Department of Civil Engineering and Architecture,
Niigata University (8050 Igarashi 2-Nocho, Niigata, 950-2181, JAPAN)
E-mail: abe@eng.niigata-u.ac.jp

In the present paper, we investigate theoretically and experimentally the number of non-zero matrix entries generated by the wavelet BEM with the Beylkin-type compression algorithm. The Beylkin-type algorithm, which is based on a prescribed *level-independent* threshold, retains the asymptotic convergence rate of BE solutions, like widely-used *level-dependent* compression schemes. The coefficient matrix compressed by the Beylkin-type scheme has $O(N^{1+\gamma})$ ($0 < \gamma < 1$, N: degree of freedom (DOF)) non-zero entries; level-dependent schemes enable us to reduce the matrix entries up to $O(N(\log N)^{\alpha})$ ($\alpha \geq 1$). However, for matrix compression using the Beylkin-type scheme the compression rate is greater than or comparable to that of the Schneider's level-dependent scheme, in the moderate DOF range.

Key Words : *wavelet BEM, matrix compression scheme, large-scale problems*

1. INTRODUCTION

In recent years, one of the major interests of boundary element (BE) researchers is the development of numerical techniques which enhance the performance of the boundary element method (BEM) in large-scale problems. The use of wavelets (wavelet BEM) is accepted as a fast solution comparable to the fast multipole BEM (FMBEM). In the wavelet BEM, wavelets are employed as the basis functions to derive a system of algebraic equations from the boundary integral equation. Many coefficients of the algebraic equations, which are defined as the double integrals including the fundamental solutions and the basis on boundary, have small values owing to the vanishing moment property of the wavelets. Hence, we will be able to truncate these small coefficients, and consequently make sparse matrices of boundary element equations. The sparse coefficient matrix enables us to reduce the computational cost of BE analysis.

As mentioned above, the key technique for the reduction of computational cost with the wavelet BEM is the truncation of small matrix entries. The choice of the truncation schemes thus influences the performance of matrix compression in wavelet-based BE analysis. The ideal situation in the analysis is to truncate as many values as possible. However, the truncation should not deteriorate the BE solution. Widely-used truncation schemes, which ensure asymptotic convergence rate of the BE solution, are classified to two types of schemes: *level-dependent* and *Beylkin-type* schemes. The main difference of these schemes is found in the way of determining the threshold of truncation.

Level-dependent truncation schemes select the truncated entries by comparing the given threshold with the distance between the supports of two basis functions. The threshold for truncation varies with the resolution level of the basis; a smaller value of threshold is used to truncate the matrix entries with a higher resolution level. This type of truncation schemes has been firstly proposed by Dahmen et al.[1)] Ref.[1)] shows the methodology for truncation in the wavelet-based BE analysis of strongly elliptic boundary value problems. Dahmen's scheme is also effective to the boundary element equation discretized by either the Galerkin- or the collocation approach using arbitrary-order wavelets. The detailed algorithms of the level-dependent schemes have been described by Schneider[2].

Other level-dependent schemes including discussions on their performances have been presented by Schwab's group[3),4),5)], Amaratunga and Castrillon-Candas[6], and Rathsfeld[7]. Schwab's schemes are similar to Schneider's one; they have constructed the algorithms by limiting the order of vanishing moments of wavelets to $M + 1$ (M: order of interpolation). This is because the choice of an extremely large order of the vanishing moments tends to increase the computational work. In Ref.[3], von Petersdorff and Schwab have discussed the matrix compression in 2-D analysis in which the integral equation of the first kind is discretized using the semi-orthogonal wavelet with the second-order vanishing moments. A practical algorithm for the equation of the second kind in 3-D has been described in Ref.[5]; the basis of the wavelet series has been defined using multiwavelets[4] with triangular support. Amaratunga and Castrillon-Candas[6] have introduced a truncation scheme for the wavelet BEM based on the Petrov-Galerkin approach. Rathsfeld[7] has proposed a matrix compression algorithm for the collocation-wavelet BEM.

The matrix compression based on the level-dependent truncation schemes except for Rathsfeld's one ensures the number of the non-zero entries of the coefficient matrix to be $O(N(\log N)^\alpha)$ ($\alpha \geq 1$). These schemes, however, have several restrictions in the application to BE analysis. The restrictions are concerned with the kind of integral equation and discretization conditions. We cannot consequently apply the level-dependent schemes to any types of elliptic boundary value problems.

On the other hand, the Beylkin-type schemes are based on the truncation with a threshold independent of the resolution level of the basis. The threshold is defined as the lower bound of the stored matrix entries in magnitude. This type of truncation scheme has been proposed by Beylkin et al. [9] They however, did not consider retaining the quality of the BE solution in the development of the truncation scheme. Numerical results, presented in Refs.[8] and [10], have been obtained by compression of the matrix under a few thresholding values. In the earlier stage of the development of wavelet BEM, these schemes were used for matrix compression and its threshold had to be determined empirically, as shown in Ref.[11]. This is because we did not have an appropriate way for determining the threshold. We can now overcome this difficulty by using the semi-analytical technique proposed by the authors[12]. The applicability of our determination strategy is independent of e.g., the kind of Fredholm integral equations and the kind of wavelets. The determined threshold enables us to compress the coefficient matrix without noticeable accuracy deterioration, even if the Beylkin-type algorithms are adopted as the truncation scheme.

The main advantages of the Beylkin-type schemes are the simple structure of the algorithms and few restrictions on implementation. The matrix compression based on these schemes however, leads to only $O(N^{1+\gamma})$ $(0 < \gamma < 1)$ non-zero entries. On the basis of the theoretical estimation of the number of the non-zero entries, several researchers have concluded that the use of the Beylkin-type scheme is of disadvantage in computational cost in contrast to the level dependent schemes. This estimation shows the DOF dependency of the storage requirement of the compressed matrix. We, however, cannot estimate the *actual* number of the entries through theoretical approaches. Previous works on wavelet-based BE analysis have been devoted to the verification of theoretical estimation or discussion of the effectiveness of wavelet BEM to reduce the computational cost of conventional BEM. The difference between the performance of the main two schemes in actual BE analysis has thus never been discussed. Moreover, the estimation on the number of non-zero entries for Beylkin-type truncation is limited to 2-D problems[12]. In our previous paper, we have never discussed the efficiency of the Beylkin-type compression in 3-D problems.

In the present work, we discuss matrix compression in wavelet-based BE analysis for 2-D and 3-D Laplace problems. The truncation of small matrix entries is based on either Schneider's level-dependent or the Beylkin-type schemes. The truncation threshold for the Beylkin-type scheme is determined with our semi-analytical technique[12]. The theoretical estimate claims that the performance of this truncation may be inferior to that of the level-dependent scheme. We thus investigate the actual compression rate of the coefficient matrix for the Beylkin-type truncation, and discuss the differences between the performance of the both compression schemes. Furthermore, the asymptotic order of the storage requirement is estimated for the Beylkin-type compression in 3-D problems. The theoretical estimation is verified through several numerical results.

2. BOUNDARY ELEMENT FORMULATION USING WAVELETS

In the present paper, we discuss 2-D and 3-D Laplace problems. The boundary integral equation for Laplace problem is formulated through a direct approach as follows:

$$c(\mathbf{x})u(\mathbf{x}) + \int_\Gamma q^*(\mathbf{x},\mathbf{y})u(\mathbf{y})d\Gamma_y = \int_\Gamma u^*(\mathbf{x},\mathbf{y})q(\mathbf{y})d\Gamma_y + f(\mathbf{x}), \tag{1}$$

where the function $f(\mathbf{x})$ is defined by either 0 for internal domain or $U_\infty(\mathbf{x})$ for external problems. The function U_∞ is the potential at the infinity. Γ denotes the boundary, and the points $\mathbf{x}, \mathbf{y}$ are on Γ. u is the potential on Γ, and the flux q is defined as $q = \partial u/\partial n$, where n stands for the outward normal direction on Γ. c is the free term.

To derive a system of algebraic equations from Eq.(1), we now approximate the true solution u and q with functions $\tilde{u}$ and $\tilde{q}$. The approximation $\tilde{u}$ and $\tilde{q}$ are defined by the following wavelet series:

$$\tilde{u} := \sum_{l=1}^{n_s} \hat{u}_{0,l}\phi_{0,l} + \sum_{k=0}^{m}\sum_{l=1}^{n_k} \tilde{u}_{k,l}\psi_{k,l}, \qquad \tilde{q} := \sum_{l=1}^{n_s} \hat{q}_{0,l}\phi_{0,l} + \sum_{k=0}^{m}\sum_{l=1}^{n_k} \tilde{q}_{k,l}\psi_{k,l}. \tag{2}$$

In Eq.(2), $\phi_{0,l}$ and $\psi_{k,l}$ are the scaling function and the wavelet function with level k, respectively. $\hat{u}_{0,l}$, $\tilde{u}_{k,l}$, $\hat{q}_{0,l}$ and $\tilde{q}_{k,l}$ are the expansion coefficients of the wavelet series. m is the finest level. n_s is the number of scaling functions $\phi_{0,l}$, and n_k corresponds to the number of wavelets $\psi_{k,l}$ with level k.

Substituting Eq.(2) into Eq.(1), we obtain the following linear algebraic equation through the Galerkin formulation:

$$\mathbf{Hu} = \mathbf{Gq} + \mathbf{f}. \tag{3}$$

In Eq.(3), $\mathbf{u}$, $\mathbf{q} \in \mathbb{R}^N$, where $\mathbb{R}$ is the set of the real numbers and N is the degree of freedom (DOF), are vectors whose components are the expansion coefficients in Eq.(2). The matrices $\mathbf{G}$, $\mathbf{H} \in \mathbb{R}^{N\times N}$ have the entries g_{ij} and h_{ij} expressed as

$$g_{ij} := \int_\Gamma w_i \int_\Gamma u^* w_j d\Gamma^2, \qquad h_{ij} := \frac{1}{2}\int_\Gamma w_i w_j d\Gamma + \int_\Gamma w_i \int_\Gamma q^* w_j d\Gamma^2, \tag{4}$$

where $i, j = 1, \ldots, N$. Both functions w_i and w_j are defined as either $\phi_{0,l}$ or $\psi_{k,l}$. Moreover, the vector $\mathbf{f} \in \mathbb{R}^N$ has the components f_i as

$$f_i = \int_\Gamma w_i(\mathbf{x}) f(\mathbf{x})\, d\Gamma. \tag{5}$$

We separate the known and unknown components of $\mathbf{u}$ and $\mathbf{q}$, and then rearrange Eq.(3). The following algebraic equation is consequently obtained:

$$\mathbf{AX} = \mathbf{b}, \tag{6}$$

where $\mathbf{b}$ is the known vector. $\mathbf{A}$ is the coefficient matrix corresponding to the unknown vector $\mathbf{X}$.

3. TRUNCATION SCHEMES FOR MATRIX COMPRESSION

The main objective of the present work is to investigate the performance of the Beylkin-type algorithms in actual BE analysis. The following discussion is based on comparison of the compression rate obtained by either Beylkin-type or level-dependent truncation. In the present work, the truncation is restricted to the matrix entries where the two basis functions w_i and w_j in Eq.(4) are both given by the wavelets ψ. Schneider's algorithm[2] is used as a level-dependent scheme.

(1) Schneider's level-dependent scheme

Schneider's truncation scheme has the distance-based truncation criterion. The truncated entries, which are chosen before calculation of Eq.(4), are associated with the basis functions satisfying the following geometrical condition:

$$\bar{r} > \delta_{k_i,k_j}, \tag{7}$$

where $\bar{r}$ is the distance between the supports of the two basis functions. δ_{k_i,k_j} is the threshold for truncation, and is defined by

$$\delta_{k_i,k_j} := a \cdot \max\left\{2^{-\min(k_i,k_j)}, (m+1)^{\frac{1}{2n+r}} 2^{(m+1)-(k_i+k_j)}\right\}, \tag{8}$$

for $p+1 = n+r$, and

$$\delta_{k_i,k_j} := a \cdot \max\left\{2^{-\min(k_i,k_j)}, 2^{\frac{(m+1)(2p'-r)-(k_i+k_j)(n+p')}{2n+r}}\right\}, \tag{9}$$

for $p+1 < n+r$. In Eqs.(8) and (9), p is the degree of the polynomials which form the wavelets. n is the order of the vanishing moments, which are defined by

$$\int_{-\infty}^{\infty} \psi(\xi)\xi^\lambda d\xi = 0, \quad (\lambda = 0, 1, \ldots, n-1), \tag{10}$$

for univariable wavelets, and

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \psi(\xi_1,\xi_2)\xi_1^{\lambda_1}\xi_2^{\lambda_2} d\xi_1 d\xi_2 = 0, \quad (\lambda_1,\ \lambda_2 \geq 0, \quad \lambda_1+\lambda_2 = 0, 1, \ldots, n-1), \tag{11}$$

for surface wavelets for 3-D BE analysis. Moreover, r is the order of a boundary integral operator: $r = -1$ for the matrix $\mathbf{G}$, while $r = 0$ for the matrix $\mathbf{H}$ in Eq.(3). The parameter p' satisfies $p + 1 < p' < n + r$, and $a > 1$. k_i and k_j are the resolution levels of the basis w_i and w_j, respectively. The truncation based on Schneider's algorithm enables us to compress the coefficient matrix at $O(N \log N)^\alpha$ ($\alpha \geq 1$) storage requirement, as shown in Ref.[2)] or will be shown in Section 4. The quasi-linear complexity of Schneider's matrix compression is enhanced to $O(N)$ by the *second compression*[2)]. This compression is based on the additional truncation of the small entries with larger difference of the levels $|k_i - k_j|$. We however, skip the second compression and thus employ the truncation algorithm only with criterion (7) and threshold (8) and (9). This is because the goal of the present work is to clarify the basic property of the two kinds of matrix compression schemes.

(2) Beylkin-type truncation scheme

In the present work, the compression algorithm proposed in Ref.[12)] is used as the Beylkin-type truncation scheme. This algorithm enables us to truncate small matrix entries without deterioration of the quality of BE solution. The first step of this compression process is to calculate the approximations $\bar{g}_{ij}$ and $\bar{h}_{ij}$ corresponding to the coefficients g_{ij} and h_{ij}. The approximations can be calculated by asymptotic expansion at the barycenter of the supports[12)]. Truncation is hence carried out before calculation of Eq.(4).

In this algorithm, the truncated entries in the coefficient matrix are selected in accordance with the following criterion:

$$\bar{g}_{ij} < \tau \cdot G_{max}, \quad \bar{h}_{ij} < \tau \cdot H_{max}. \tag{12}$$

In assembly of the coefficient matrix, the entries which satisfy Eq.(12) are truncated, if the distance $\bar{r}$ satisfies $\bar{r} \geq \nu(2^{-k_i} + 2^{-k_j})$ ($\nu > 0$). The entries where the distance between the supports of w_i and w_j are smaller than the distance $\nu(2^{-k_i} + 2^{-k_j})$ are stored without truncation. This exception is introduced to avoid the truncation of matrix entries with overlapping supports. In Eq.(12), G_{max} and H_{max} are the maxima of the coefficients g_{ij} and h_{ij}, respectively. τ is the threshold for truncation, and is predetermined by the semi-analytical technique proposed in Ref.[12)]. The threshold is then set to a level where a larger number of coefficients are truncated without accuracy loss. Such a threshold can be determined by allowing the truncation-induced error upto the level of discretization error of BE solution. The semi-analytical technique requires error estimation of BE solution; the discretization error is indirectly estimated by the residual of the boundary integral equation. The threshold τ can be consequently approximated by

$$\tau \approx \frac{1}{2 \cdot \|\mathbf{H}\|} \cdot \frac{\|\mathbf{c}\|}{\|\mathbf{u}\|}. \tag{13}$$

In Eq.(13), $\mathbf{H}$ and $\mathbf{u}$ are the matrix and vector in Eq.(3). The components c_i ($i = 1, 2, \ldots, N$) of the vector $\mathbf{c}$ are defined as

$$\begin{aligned} c_i := -\frac{1}{2} \int_\Gamma w_i(\hat{\hat{u}} - \tilde{u}) d\Gamma \\ - \int_\Gamma w_i \int_\Gamma q^*(\hat{\hat{u}} - \tilde{u}) d\Gamma^2, \end{aligned} \tag{14}$$

where $\hat{\hat{u}}$ is the wavelet series of the true solution u. In the determination process, $\hat{\hat{u}}$ is given by the wavelet series of the higher-order interpolation of BE solution $\tilde{u}$. This is because the true solution u is unknown. Moreover, the norms $\|\mathbf{H}\|$ and $\|\mathbf{c}\|/\|\mathbf{u}\|$ are for the BE solution with the same DOF N as that of main analysis. This calculation is only for setting the threshold τ but will require a huge computational cost. The norms in Eq.(13) are approximately predicted as follows. The norm $\|\mathbf{c}\|/\|\mathbf{u}\|$ is estimated by

$$\frac{\|\mathbf{c}\|}{\|\mathbf{u}\|} \approx \alpha' \cdot N^{-\beta}, \tag{15}$$

where the parameters α' and β are independent of DOF N. In the present technique these parameters are set on the basis of auxiliary BE analysis with sufficiently small DOF. The norm $\|\mathbf{H}\|$ is approximated by the norm of the matrix $\tilde{\mathbf{H}}$ obtained through an auxiliary analysis.

The threshold τ for the Beylkin-type compression is hence determined as

$$\tau = \frac{\alpha' N^{-\beta}}{2 \cdot \|\tilde{\mathbf{H}}\|}. \tag{16}$$

The truncation using the threshold (16) retains the accuracy of BE solution, as reported in Ref.[12)].

The original algorithm proposed in Ref.[12)] indeed includes also the *a posteriori* truncation which is carried out after the calculation of matrix coefficients. We however, consider the *a priori* truncation which is implemented before the calculation for matrix assembly. This is because the estimation of the number of stored matrix entries to be presented in the following will correspond to the *a priori* truncation.

4. ASYMPTOTIC ESTIMATION OF THE NUMBER OF NON-ZERO MATRIX ENTRIES

In the present section, we show a mathematical approach for estimating the number of stored entries of the matrix compressed with the above truncation schemes. The coefficient matrix in wavelet BE analysis is defined as an assembly of the blocks specified with the resolution level $(k_i,\, k_j)$. The total size of storage can be thus estimated by summing the number of non-zero entries in every block. The framework of this estimation has been presented by von Petersdorff and Schwab[3] and Lage and Schwab[5]; they have discussed the complexity of BE analysis based on Schwab's level-dependent matrix compression.

We first denote by $\mathbf{B}$ either the matrices $\mathbf{G}$ or $\mathbf{H}$ in Eq.(3), and define $\mathbf{B}_{k_i,k_j}$ as the block with the level $(k_i,\, k_j)$. The storage requirement $\mathscr{N}(\mathbf{B}_{k_i,k_j})$ of the block $\mathbf{B}_{k_i,k_j}$ is bounded due to truncation of small entries as

$$\mathscr{N}(\mathbf{B}_{k_i,k_j}) \le C(2^{k_i s} + 2^{k_j s} + \gamma 2^{(k_i+k_j)s}\delta^s), \tag{17}$$

where $s = 1$ for 2-D, and $s = 2$ for 3-D. The constant γ satisfies $\gamma > 0$. δ is the supremum of the distance between the supports of the wavelets which are used to generate the stored matrix entries.

(1) Schneider's level-dependent scheme

Schneider's truncation scheme has the distance-based threshold as shown in the previous section. The supremum δ in Eq.(17) is determined to be δ_{k_i,k_j} given by either Eq.(8) or (9) depending on p, n and r.

a) $p + 1 = n + r$

The supremum δ in Eq.(17) is given by Eq.(8) for $p + 1 = n + r$, and is set to $\delta = (m+1)^{\frac{1}{2n+r}} 2^{(m+1)-(k_i+k_j)}$ at every level $(k_i,\, k_j)$ for the finest level $m \ge 1$. In wavelet BEM, remarkable reduction of computational cost is achieved by truncating the wavelet series having sufficiently large finest level m for approximation. The following estimation is thus developed under the assumption of $m \ge 1$. The number of entries of the matrix $\mathbf{B}$, $\mathscr{N}(\mathbf{B})$, is bounded as follows:

$$\begin{aligned}\mathscr{N}(\mathbf{B}) &\le C' \sum_{k_i=0}^{m}\sum_{k_j=0}^{m} \mathscr{N}(\mathbf{B}_{k_i,k_j}) \\ &\le C_1''(m+1)2^{(m+1)s} \\ &\qquad + C_2'' 2^{(m+1)s}(m+1)^{2+\frac{s}{2n+r}} \\ &\le C_1' N\log N + C_2' N(\log N)^{2+\frac{s}{2n+r}},\end{aligned} \tag{18}$$

where $2^{(m+1)s} \approx N$ and $(m+1) \approx \log N$. The storage size $\mathscr{N}(\mathbf{B})$ is hence estimated to be $O(N(\log N)^{2+\frac{s}{2n+r}})$. The number of entries in the matrix $\mathbf{G}$ or $\mathbf{H}$ can be easily estimated from Eq.(18) as one sets the order r of an integral operator to $r = -1$ for G-matrix or $r = 0$ for H-matrix.

b) $p + 1 < n + r$

For $p + 1 < n + r$, the supremum δ in Eq.(17) is given by Eq.(9). The threshold δ_{k_i,k_j} of Eq.(9) is set to either the first- or second term of the right-hand side in accordance with the level $(k_i,\, k_j)$.

We now consider the level at which the first term of Eq.(9) is chosen as the threshold. For $k_i \ge k_j$, the blocks with the threshold $\delta = 2^{-\min(k_i,k_j)}$ have the level $(k_i,\, k_j)$ satisfying

$$\begin{aligned}k_j &< \sigma k_i - \rho, \\ \sigma &:= \frac{n+p'}{n+r-p'} > 0, \\ \rho &:= \frac{(2p'-r)(m+1)}{n+r-p'}.\end{aligned} \tag{19}$$

The number of non-zero entries in the corresponding block $\mathbf{B}_{k_i,k_j}$ is then estimated as $\mathscr{N}(\mathbf{B}_{k_i,k_j}) \le C_1(2^{k_i s} + 2^{k_j s})$. On the other hand, the block where truncation threshold is set to the second term of Eq.(9) has the number of entries:

$$\begin{aligned}\mathscr{N}(\mathbf{B}_{k_i,k_j}) \le C_2(&2^{k_i s} + 2^{k_j s} \\ &+ 2^{\kappa s(m+1)} \cdot 2^{\kappa' s(k_i+k_j)}),\end{aligned} \tag{20}$$

where $\kappa := (2p'-r)/(2n+r)$ and $\kappa' := (n+r-p')/(2n+r)$.

Eq.(19) shows that the blocks with the level $k_i < \rho/\sigma$ are always compressed using the threshold defined by the second term of Eq.(9). Since the threshold δ_{k_i,k_j} for Schneider's truncation scheme satisfies $\delta_{k_i,k_j} = \delta_{k_j,k_i}$, the number of non-zero entries of the matrix $\mathbf{B}$ is estimated by

$$\begin{aligned}\mathscr{N}(\mathbf{B}) \le\ & 2C_1 \sum_{k_i=\rho/\sigma}^{m}\ \sum_{k_j=0}^{\sigma k_i-\rho-1} (2^{k_i s} + 2^{k_j s}) \\ &+ 2C_2 \sum_{k_i=0}^{\rho/\sigma-1}\ \sum_{k_j=0}^{k_i} \big(2^{k_i s} + 2^{k_j s} \\ &\qquad + 2^{\kappa s(m+1)} \cdot 2^{\kappa' s(k_i+k_j)}\big) \\ &+ 2C_2 \sum_{k_i=\rho/\sigma}^{m}\ \sum_{k_j=\sigma k_i-\rho}^{k_i} \big(2^{k_i s} + 2^{k_j s} \\ &\qquad + 2^{\kappa s(m+1)} \cdot 2^{\kappa' s(k_i+k_j)}\big) \\ \le\ & C_1' N\log N + C_2' N \\ &+ C_3' N^{1-\frac{2(n+r-p')^2}{(2n+r)(n+p')}} + C_4' N^{1-\frac{n+r-p'}{n+p'}},\end{aligned} \tag{21}$$

where $2^{(m+1)s} \approx N$ and $(m+1) \approx \log N$. In Eq.(21), we have

$$1 - \frac{2(n+r-p')^2}{(2n+r)(n+p')} < 1,$$
$$1 - \frac{n+r-p'}{n+p'} < 1.$$

The number of entries, $\mathscr{N}(\mathbf{B})$, is consequently of $O(N \log N)$ for $p+1 < n+r$, as shown in Ref.[2].

(2) Beylkin-type scheme

a) Distance-based threshold

The storage requirement for Beylkin-type matrix compression discussed in the Section **3(2)** is estimated through the same technique as that for Schneider's compression. In estimation of storage size of the coefficient matrix, we first have to describe the threshold as the maximum distance between the supports of the wavelets corresponding to the stored entries. This is because original threshold (12) for the Beylkin-type scheme is defined as the maximum of the stored entries in magnitude.

We now consider the matrix entries g_{ij} and h_{ij} described in Eq.(4). On the basis of Schneider's estimation [2], we have

$$|g_{ij}| \leq C_g \frac{2^{-(n+s/2)(k_i+k_j)}}{\bar{r}^{s+2n-1}}, \quad |h_{ij}| \leq C_h \frac{2^{-(n+s/2)(k_i+k_j)}}{\bar{r}^{s+2n}}, \qquad (22)$$

where $\bar{r}$ is the distance between two basis supports. The truncated entries satisfy Eq.(12), and their maximum is calculated under the distance δ_g or δ_h between the supports of wavelets. The distance δ_g and δ_h are derived from Eqs.(12) and (22) as follows:

$$\delta_g = C'_g \tau^{-\frac{1}{s+2n-1}} 2^{-\frac{s+2n}{2(s+2n-1)}(k_i+k_j)}, \quad \delta_h = C'_h \tau^{-\frac{1}{s+2n}} 2^{-\frac{1}{2}(k_i+k_j)}. \qquad (23)$$

Moreover, the truncated entries are chosen among the matrix entries with $\bar{r} > r_0 := \nu(2^{-k_i} + 2^{-k_j})$, using Eq.(12). The truncation criterion with the distance-based threshold is hence described as

truncate the entries with $\bar{r} > \delta$ *for* $r_0 < \delta$, *while*
truncate the entries with $\bar{r} > r_0$ *for* $r_0 \geq \delta$, (24)

where the threshold δ is set to δ_g for the matrix $\mathbf{G}$ or δ_h for the matrix $\mathbf{H}$. δ_g and δ_h are defined in Eq.(23).

b) Resolution level corresponding to the blocks with $r_0 < \delta$

As shown in Eq.(24), the distance-based threshold for Beylkin-type compression is determined to be either δ or r_0 in accordance with their magnitude. The distance δ and r_0 both depend on the level $(k_i,\ k_j)$. The threshold in truncation criterion (24) is hence set at every block of the coefficient matrix.

We now consider the level $(k_i,\ k_j)$ corresponding to the blocks satisfying $r_0 < \delta$. In the blocks with $r_0 < \delta$, we have the following relation:

$$\nu(2^{-k_i} + 2^{-k_j}) < C'_g \tau^{-\frac{1}{s+2n-1}} 2^{-\frac{(s+2n)(k_i+k_j)}{2(s+2n-1)}}, \quad (\text{matrix } \mathbf{G}),$$
$$\nu(2^{-k_i} + 2^{-k_j}) < C'_h \tau^{-\frac{1}{s+2n}} 2^{-\frac{k_i+k_j}{2}}, \quad (\text{matrix } \mathbf{H}), \qquad (25)$$

Multiplying both sides of Eq.(25) by $2^{k_i+k_j}$, we have

$$2^{k_i} + 2^{k_j} < \frac{C'_g}{\nu} \tau^{-\frac{1}{s+2n-1}} 2^{(1-\frac{1}{s+2n-1})\frac{k_i+k_j}{2}},$$
$$2^{k_i} + 2^{k_j} < \frac{C'_h}{\nu} \tau^{-\frac{1}{s+2n}} 2^{\frac{k_i+k_j}{2}}, \qquad (26)$$

where $s+2n-1 > 0$ and $1 - 1/(s+2n-1) < 1$. Hence, the resolution level $(k_i,\ k_j)$ satisfying the first inequality of Eq.(26) also satisfies the following relation:

$$2^{k_i} + 2^{k_j} < \frac{C'_g}{\nu} \tau^{-\frac{1}{s+2n-1}} 2^{\frac{k_i+k_j}{2}}. \qquad (27)$$

Replacing the first inequality of Eq.(26) with Eq.(27), we have

$$2^{k_j} - \frac{C'_g}{\nu} \tau^{-\frac{1}{s+2n-1}} 2^{\frac{k_i+k_j}{2}} + 2^{k_i} < 0, \quad (\text{matrix } \mathbf{G}),$$
$$2^{k_j} - \frac{C'_h}{\nu} \tau^{-\frac{1}{s+2n}} 2^{\frac{k_i+k_j}{2}} + 2^{k_i} < 0, \quad (\text{matrix } \mathbf{H}). \qquad (28)$$

We now define $x = 2^{k_j/2}$ and $t = 2^{k_i/2}$ in Eq.(28), and introduce a new constant $\mu = C'_g \tau^{-\frac{1}{s+2n-1}}$ for matrix $\mathbf{G}$ or $\mu = C'_h \tau^{-\frac{1}{s+2n}}$ for matrix $\mathbf{H}$. Eq.(28) is thus rewritten as

$$x^2 - \frac{\mu}{\nu} xt + t^2 < 0. \qquad (29)$$

The bounds of x which satisfy Eq.(29) are easily calculated as

$$x = \frac{t}{2} \left(\frac{\mu}{\nu} \pm \sqrt{\frac{\mu^2}{\nu^2} - 4} \right). \qquad (30)$$

Assuming $\mu/\nu \gg 2$, we can approximate the second term of Eq.(30) as

$$\frac{t}{2} \sqrt{\frac{\mu^2}{\nu^2} - 4} \approx \frac{t}{2} \left(\frac{\mu}{\nu} - \frac{2\nu}{\mu} \right). \qquad (31)$$

On the basis of Eq.(31), Eq.(30) is described as follows:

$$x \approx \frac{\mu}{\nu} t, \quad \frac{\nu}{\mu} t. \qquad (32)$$

We substitute $\mu/\nu = 2^{\omega/2}$ ($\omega > 0$), $x = 2^{k_j/2}$ and $t = 2^{k_i/2}$ into the solution of Eq.(29), and consider the logarithmic expression of the resulting inequalities. The level (k_i, k_j) satisfying $r_0 < \delta$ is consequently specified with

$$\begin{aligned} &k_i - \omega_g < k_j < k_i + \omega_g, && \text{(matrix } \mathbf{G}\text{)}, \\ &k_i - \omega_h < k_j < k_i + \omega_h, && \text{(matrix } \mathbf{H}\text{)}, \end{aligned} \tag{33}$$

where the constants ω_g and ω_h are defined as follows:

$$2^{\frac{\omega_g}{2}} = \frac{C'_g}{\nu}\tau^{-\frac{1}{s+2n-1}}, \quad 2^{\frac{\omega_h}{2}} = \frac{C'_h}{\nu}\tau^{-\frac{1}{s+2n}}. \tag{34}$$

c) Number of stored entries of the blocks $\mathbf{G}_{k_i,k_j}$ and $\mathbf{H}_{k_i,k_j}$

The number of the entries in the compressed blocks of coefficient matrices is estimated using Eq.(17), as in the estimation for Schneider's truncation scheme. We now define the blocks of the matrices $\mathbf{G}$ and $\mathbf{H}$ with the level (k_i, k_j) as $\mathbf{G}_{k_i,k_j}$ and $\mathbf{H}_{k_i,k_j}$. In the Beylkin-type compression algorithm shown in Eq.(24), the threshold is determined to δ_g or δ_h defined in Eq.(23), or $r_0 = \nu(2^{-k_i} + 2^{-k_j})$. The choice of threshold is based on the level (k_i, k_j) and the kind of the matrix. The storage size $\mathscr{N}(\mathbf{G}_{k_i,k_j})$ and $\mathscr{N}(\mathbf{H}_{k_i,k_j})$ are hence bounded as follows:

$$\mathscr{N}(\mathbf{G}_{k_i,k_j}) \le \begin{cases} C_{g1}\Big(2^{k_i s} + 2^{k_j s} \\ \qquad +\gamma'\tau^{-\frac{s}{s+2n-1}}2^{\frac{k_i+k_j}{2}\varepsilon s}\Big), \\ \qquad\qquad (r_0 < \delta) \\ C'_{g1}(2^{k_i s} + 2^{k_j s}), \\ \qquad\qquad (r_0 \ge \delta) \end{cases} \tag{35}$$

$$\mathscr{N}(\mathbf{H}_{k_i,k_j}) \le \begin{cases} C_{h1}\Big(2^{k_i s} + 2^{k_j s} \\ \qquad +\gamma'\tau^{-\frac{s}{s+2n}}2^{\frac{k_i+k_j}{2}s}\Big), \\ \qquad\qquad (r_0 < \delta) \\ C'_{h1}(2^{k_i s} + 2^{k_j s}), \\ \qquad\qquad (r_0 \ge \delta) \end{cases} \tag{36}$$

where $\gamma' > 0$ and $\varepsilon := 2 - (s + 2n)/(s + 2n - 1)$.

d) Number of non-zero entries of matrix G

We now estimate the number of stored entries in the matrix $\mathbf{G}$. The storage requirement $\mathscr{N}(\mathbf{G})$ of the matrix $\mathbf{G}$ can be estimated by summing the number of entries in every block $\mathbf{G}_{k_i,k_j}$. The number of entries in the block $\mathbf{G}_{k_i,k_j}$, $\mathscr{N}(\mathbf{G}_{k_i,k_j})$, has been presented in Eq.(35). The threshold δ_g defined in Eq.(23) is invariant under the exchange of k_i and k_j. The storage requirement $\mathscr{N}(\mathbf{G})$ is hence bounded as follows:

$$\begin{aligned} \mathscr{N}(\mathbf{G}) &\le C\sum_{k_i=0}^{m}\sum_{k_j=0}^{m}\mathscr{N}(\mathbf{G}_{k_i,k_j}) \\ &\le C'_1\sum_{k_i=0}^{m}\sum_{k_j=0}^{m}(2^{k_i s} + 2^{k_j s}) \\ &\quad + C'_2\tau^{\frac{-s}{s+2n-1}}\sum_{k_i=0}^{m-\omega_g}\sum_{k_j=k_i}^{k_i+\omega_g}2^{\frac{\lambda s}{2}(k_i+k_j)} \\ &\quad + C'_3\tau^{\frac{-s}{s+2n-1}}\sum_{k_i=m-\omega_g+1}^{m}\sum_{k_j=k_i}^{m}2^{\frac{\lambda s}{2}(k_i+k_j)}, \end{aligned} \tag{37}$$

where $\lambda = (s + 2n - 2)/(s + 2n - 1)$.

The summations in the terms of Eq.(37) are calculated as

$$\sum_{k_i=0}^{m}\sum_{k_j=0}^{m}(2^{k_i s} + 2^{k_j s}) \le \bar{C}_1(m+1)2^{(m+1)s}, \tag{38}$$

$$\begin{aligned} &\sum_{k_i=0}^{m-\omega_g}\sum_{k_j=k_i}^{k_i+\omega_g}2^{\frac{\lambda s}{2}(k_i+k_j)} \\ &\qquad \le \bar{C}''_2\tau^{\frac{\lambda s}{s+2n-1}}2^{\lambda s(m+1)}, \end{aligned} \tag{39}$$

$$\begin{aligned} &\sum_{k_i=m-\omega_g-1}^{m}\sum_{k_j=k_i}^{m}2^{\frac{\lambda s}{2}(k_i+k_j)} \\ &\qquad \le \bar{C}'_3 2^{\lambda s(m+1)}\left(1 + \tau^{\frac{\lambda s}{s+2n-1}} + \tau^{\frac{2\lambda s}{s+2n-1}}\right). \end{aligned} \tag{40}$$

Substituting Eqs.(38)–(40) into Eq.(37), we have

$$\begin{aligned} \mathscr{N}(\mathbf{G}) \le C_4 N\log N + N^{\lambda}\Big(&C_5 N^{\frac{s\beta}{s+2n-1}} \\ &+ C_6 N^{\frac{(1-\lambda)s\beta}{s+2n-1}} + C_7 N^{\frac{(1-2\lambda)s\beta}{s+2n-1}}\Big). \end{aligned} \tag{41}$$

Note that $\tau = \alpha' N^{-\beta}$, $m + 1 \approx \log N$ and $2^{s(m+1)} \approx N$.

In Eq.(41), $\lambda = 1 - 1/(s + 2n - 1)$ satisfies $0 < \lambda < 1$ with $s + 2n - 1 > 0$. The indices of the terms in Eq.(41) thus have the following relation:

$$\frac{(1-2\lambda)s\beta}{s+2n-1} < \frac{(1-\lambda)s\beta}{s+2n-1} < \frac{s\beta}{s+2n-1}. \tag{42}$$

Rearranging Eq.(41) on the basis of Eq.(42), we have

$$\begin{aligned} \mathscr{N}(\mathbf{G}) &\le C'_4 N\log N + C''_5 N^{\lambda+\frac{s\beta}{s+2n-1}} \\ &= C'_4 N\log N + C''_5 N^{1+\frac{s\beta-1}{s+2n-1}}. \end{aligned} \tag{43}$$

Therefore, the number of stored entries in the matrix $\mathbf{G}$ shows either $O(N\log N)$ for $\beta \le 1/s$ or $O(N^{1+(s\beta-1)/(s+2n-1)})$ for $\beta > 1/s$.

e) Number of non-zero entries of matrix H

The number of stored entries of matrix $\mathbf{H}$, $\mathscr{N}(\mathbf{H})$, can be estimated through the same technique as that for the matrix $\mathbf{G}$. Summing the storage requirement of every block $\mathbf{H}_{k_i,k_j}$ in accordance with Eqs.(33) and (36), we have

$$\begin{aligned}\mathscr{N}(\mathbf{H}) &\le C\sum_{k_i=0}^{m}\sum_{k_j=0}^{m}\mathscr{N}(\mathbf{H}_{k_i,k_j})\\ &\le C_1'\sum_{k_i=0}^{m}\sum_{k_j=0}^{m}(2^{k_i s}+2^{k_j s})\\ &+C_2'\tau^{\frac{-s}{s+2n}}\sum_{k_i=0}^{m-\omega_h}\sum_{k_j=k_i}^{k_i+\omega_h}2^{\frac{s(k_i+k_j)}{2}}\\ &+C_3'\tau^{\frac{-s}{s+2n}}\sum_{k_i=m-\omega_h+1}^{m}\sum_{k_j=k_i}^{m}2^{\frac{s(k_i+k_j)}{2}}.\end{aligned} \tag{44}$$

The summations in the terms of Eq.(44) are calculated as follows:

$$\sum_{k_i=0}^{m-\omega_h}\sum_{k_j=k_i}^{k_i+\omega_h}2^{\frac{k_i+k_j}{2}s}\le \bar{C}_1\tau^{\frac{s}{s+2n}}2^{s(m+1)}, \tag{45}$$

$$\begin{aligned}\sum_{k_i=m-\omega_h+1}^{m}\sum_{k_j=k_i}^{m}2^{\frac{k_i+k_j}{2}s}&\le \bar{C}_2 2^{s(m+1)}\\ &+\bar{C}_3\tau^{\frac{s}{s+2n}}2^{s(m+1)}+\bar{C}_4\tau^{\frac{2s}{s+2n}}2^{s(m+1)}.\end{aligned} \tag{46}$$

Substituting Eqs.(38), (45) and (46) into Eq.(44), we have the following estimation:

$$\begin{aligned}\mathscr{N}(\mathbf{H}) &\le C_4 N\log N + C_5 N\\ &+C_6 N^{1-\frac{s\beta}{s+2n}}+C_7 N^{1+\frac{s\beta}{s+2n}},\end{aligned} \tag{47}$$

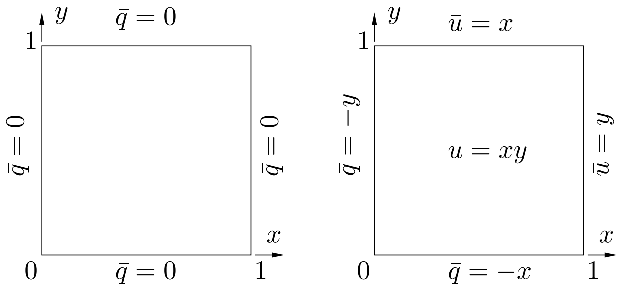
where $\tau=\alpha' N^{-\beta}$, $\log N\approx m+1$, $N\approx 2^{s(m+1)}$ and $s\beta/(s+2n)>0$. The storage requirement of the matrix $\mathbf{H}$ hence is of $O(N^{1+s\beta/(s+2n)})$.

5. NUMERICAL RESULTS

(1) Problem description

The numerical tests were undertaken to investigate the performance of the Beylkin-type truncation scheme in actual BE analysis. The test examples were Laplace problems of 2-D or 3-D. The matrix compression in the present tests was carried out using either the Beylkin-type or Schneider's truncation schemes; the detail of these truncation algorithms has been described in Section **3**. The results obtained through the Beylkin-type matrix compression will be shown with the label "Beylkin" in the present section. Schneider's truncation algorithm was implemented with $a=1$. In the figures in the present section, the label "Case 1" indicates the results obtained by compression with threshold (8). The results for truncation with threshold (9) are labelled "Case 2".

The present 2-D problems have two kinds of boundary conditions. The first example is a Neumann problem with an external domain shown in **Fig. 1**(a). The Neumann condition is set to $\bar{q}=0$ on the boundary, and the potential at the infinity shows $U_\infty = x$. The second example of 2-D is a mixed boundary value problem with an internal domain. The details of boundary conditions and true solution are illustrated in **Fig. 1**(b).



(a) External Neumann problem. ($U_\infty = x$)

(b) Mixed boundary value problem for an internal domain.

Fig.1 Test examples for 2-D analysis.

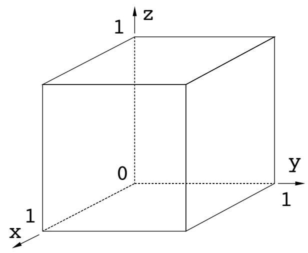


Fig.2 Test example for 3-D analysis. (Neumann problem for an external domain. Boundary conditions: $\bar{q}=0$, $U_\infty = x$)

In the present work, we also deal with 3-D problem. The test example for 3-D is an external Neumann problem. **Fig. 2** shows the boundary conditions of the 3-D example. The Neumann condition is set to $\bar{q}=0$ on the boundary, and U_∞ is defined as $U_\infty = x$.

(2) 2-D Neumann problems for external domain

We first present the results for the 2-D external Neumann problem shown in **Fig. 1**(a). **Fig. 3** depicts the number of non-zero entries in the coefficient matrix $\mathbf{A}$ in Eq.(6). For the present problem all the entries of the matrix $\mathbf{A}$ are the coefficients h_{ij} defined in Eq.(4). The results shown in **Fig. 3** thus correspond to the number of entries of the matrix $\mathbf{H}$. In calculation of BE solution, the approximations $\tilde{u}$ and $\tilde{q}$ were defined as the wavelet series with the Haar wavelets [11]. Using $\tilde{u}$ and $\tilde{q}$, the true solutions u and q were expanded in every sub-boundary divided at four corners of the square domain. The Haar wavelet consists of two piecewise constant functions ($p = 0$), and has the first-order vanishing moment ($n = 1$). For Haar wavelets, the boundary integral equation is thus discretized under $p + 1 = n + r$ for the matrix $\mathbf{H}$, and under $p+1 > n+r$ for the matrix $\mathbf{G}$. This fact implies that Schneider's level-dependent truncation scheme is not applicable to compression of the matrix $\mathbf{G}$. In the present problem, Schneider's scheme can be used for matrix compression without any restriction, because the assembly of the matrix $\mathbf{G}$ is skipped.

We now compare the storage requirement of the coefficient matrix compressed by the Beylkin-type algorithm, with Case 1 in Schneider's truncation. The results corresponding to Case 1 were obtained through compression with appropriate threshold (8). **Fig. 3** shows that Beylkin-type compression leads to smaller number of stored entries than that of Case 1, in the small DOF range. The difference between both storage requirements however, decreases as the DOF increases. In the range of DOF larger than 10,000, the matrix compression of higher rates is achieved by implementation of Schneider's algorithm. The superiority of Schneider's scheme with appropriate threshold in storage requirement is limited in the range where the finest level m is set to a sufficiently large integer.

As described in Section **4**, the quasi-linear complexity is ensured to Schneider's truncation alone. The difference between the storage requirements for the above compression schemes is small within the DOF range chosen in the present test. In other words, the matrix compression using the Beylkin-type scheme shows high compression rate comparable to that for Schneider's truncation with appropriate threshold (8). This fact shows that the Beylkin-type scheme is effective for the reduction of computational cost in actual BE analysis for 2-D Neumann problems.

In the present example, we set the threshold for the Beylkin-type truncation to $\tau = \alpha' N^{-1.28}$ through an auxiliary BE analysis mentioned in Section **3**. The number of stored entries of the matrix $\mathbf{A}$ is hence predicted to be increased at $O(N^{1.43})$ from our estimation; the corresponding numerical results show the behaviour of $O(N^{1.43})$. The present estimation described in the previous section is efficient to predict the asymptotical order of the storage requirement of the compressed H-matrix for 2-D.

We next compare the storage requirement for Beylkin-type matrix compression with the numerical results for Schneider's truncation under inappropriate threshold (9). Schneider's compression using threshold (9) has been classified

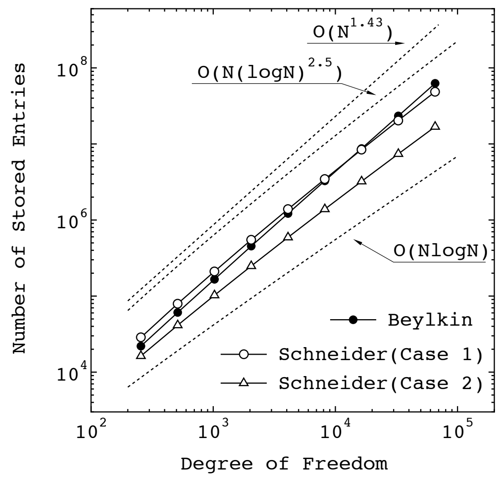


Fig.3 The number of stored entries of the coefficient matrix for the 2-D external problem shown in **Fig. 1**(a). The boundary values were approximated using the Haar wavelets ($p = 0$ and $n = 1$). The label "Case 2" indicates the results obtained through compression with inappropriate threshold (9) and $p' = 1$.

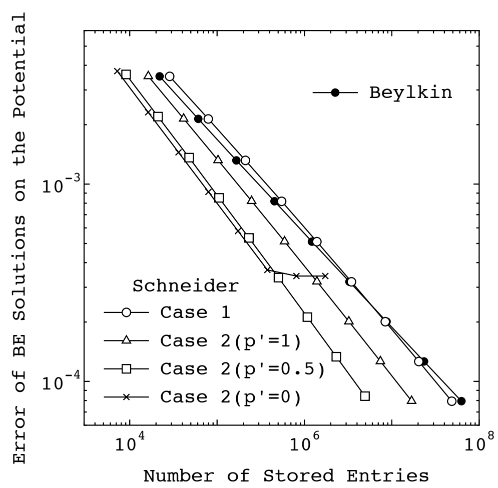


Fig.4 The storage requirement and the error of the corresponding BE solution. The test example is the 2-D external Neumann problem shown in **Fig. 1**(a). The Haar wavelets are used for the basis of the approximation $\tilde{u}$ and $\tilde{q}$. The parameter p' in the Case 2 for Schneider's truncation is set out of the appropriate range.

to the Case 2 in the present paper; the results for this case were obtained under $p' = 1$. For the Case 2 with $p' = 1$, Schneider's compression is superior in storage requirement to Beylkin-type truncation. Implementation of Schneider's scheme then does not cause accuracy deterioration of BE solution, as shown in **Fig. 4**. We thus attempted to truncate the small coefficients using threshold (9) with $p' < 1$. **Fig. 4** indicates the number of stored entries and the error of the corresponding BE solution. In compression of the matrix **A**, we set the parameter p' to 0, 0.5 and 1.

As shown in **Fig. 4**, the choice of the parameter p' to a smaller value results in the higher compression rates of the matrix **A**. Threshold (9) for Schneider's algorithm is not applicable to the present example and discretization. Schneider's scheme then does not always ensure the asymptotic convergence rate of BE solution. However, compression with $p' = 0.5$ or $p' = 1$ does not disturb the convergence of the solution in the present numerical test. For $p' = 0$, the error of BE solution reaches a lower bound at a critical value of DOF; the solution with a larger DOF than the critical value has the error comparable to the lower bound. This fact clearly results from overtruncation. A similar lower bound to that for $p' = 0$ may be found in matrix compression with $p' = 0.5$ and $p' = 1$ in larger DOF range. We thus have to carefully choose the truncation parameter p', if Schneider's algorithm with an inappropriate threshold is implemented in actual BE analysis.

(3) 2-D mixed boundary value problems for internal domain

We now discuss the performance of the Beylkin-type scheme in 2-D mixed boundary value problems. For this type of problems, the coefficient matrix **A** is assembled through the calculation and truncation of coefficients g_{ij} and h_{ij}. These coefficients were calculated under the discretization of a boundary integral equation with the piecewise constant non-orthogonal spline wavelets[11]. The wavelets used in the present numerical test have the third-order vanishing moments; $p = 0$ and $n = 3$. The inequality $p + 1 < n + r$ then holds for the coefficients in both the matrices **G** and **H**. Schneider's truncation scheme is thus implemented with threshold (9). The parameter p' can be set to a value in the range of $1 < p' < 2$.

Fig. 5 shows the number of stored entries in the matrix **A**, which is compressed with either the Beylkin-type or Schneider's algorithms. The quality of the both truncation techniques in matrix compression is similar to that for external Neumann problems presented in the previous subsection. In smaller DOF range the Beylkin-type scheme enables us to compress the matrix **A** at higher rates. The superiority of this technique may disappear as the DOF increases. The difference between the storage requirements for the above two types of compression is small in the present DOF range. Hence, the employment of the Beylkin-type scheme, which ensures $O(N^{1+\gamma})$ storage requirement, will not be a disadvantage with regard to the computational cost.

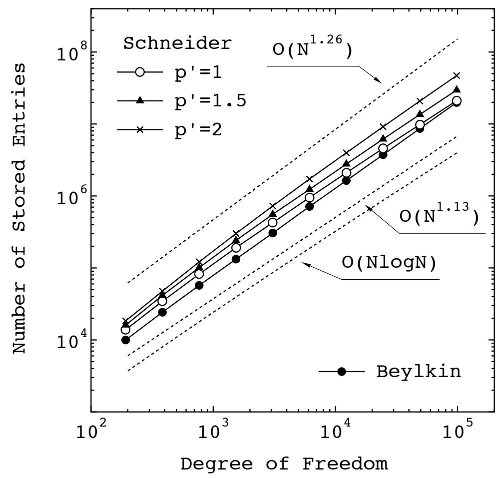


Fig.5 The number of stored entries of the coefficient matrix for the 2-D mixed boundary value problem shown in **Fig. 1**(b). The baoundary values were approximated using the piecewise constant non-orthogonal wavelets with the third-order vanishing monents ($p = 0$ and $n = 3$). The threshold for Schneider's matrix compression was determined with Eq.(9).

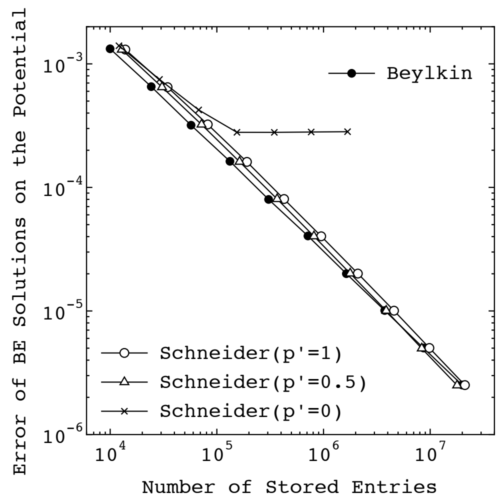


Fig.6 The storage requirement and the error of the corresponding BE solution. The test example is the 2-D mixed boundary value problem shown in **Fig. 1**(b). The boundary values were approximated using the piecewise constant non-orthogonal wavelets with the third-order vanishing monents ($p = 0$ and $n = 3$). The appropriate threshold for Schneider's compression is determined with the parameter p' in $1 < p' < 2$.

In the present example, the threshold for Beylkin-type compression was determined with $\beta = 1.78$. The number of stored entries in the matrix **G** or **H** is predicted to show $O(N^{1.13})$ for

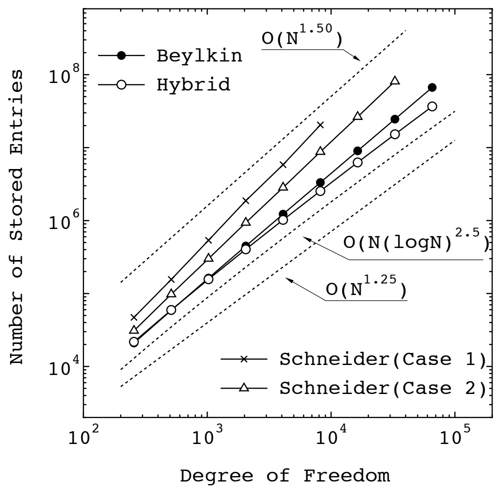


Fig.7 The number of stored entries in the coefficient matrix **A** for the 2-D problem illustrated in **Fig. 1**(b). The boundary values were approximated using the Haar wavelets. Schneider's algorithm is not applicable to compression of the matrix **G**. The results labelled "Case 2" were obtained through truncation with $p' = 1$.

G-matrix or $O(N^{1.26})$ for H-matrix; the numerical results indicate the storage requirement of $O(N^{1.21})$. The power "1.21" is within the estimated range [1.13, 1.26]. The validity of our estimation for compression of the matrices **G** and **H** is thus verified through the present numerical results.

We next consider the disadvantage caused by using the inappropriate threshold in Schneider's matrix compression. **Fig. 6** depicts the storage requirement of the coefficient matrix **A** and the error of the corresponding BE solution. The numerical results were obtained through discretization with the piecewise constant non-orthogonal wavelets; these basis functions have the third-order vanishing moments. The inappropriate threshold for Schneider's compression was examined for $p' = 0$, 0.5 and 1. As shown in **Fig. 6**, the matrix compression with $p' = 0.5$ or 1 retains the convergence rate within the present DOF range. The choice of the parameter $p' = 0$ however, results in disturbance of the convergence of BE solution. These show the same tendency in the error as that for 2-D external Neumann problems.

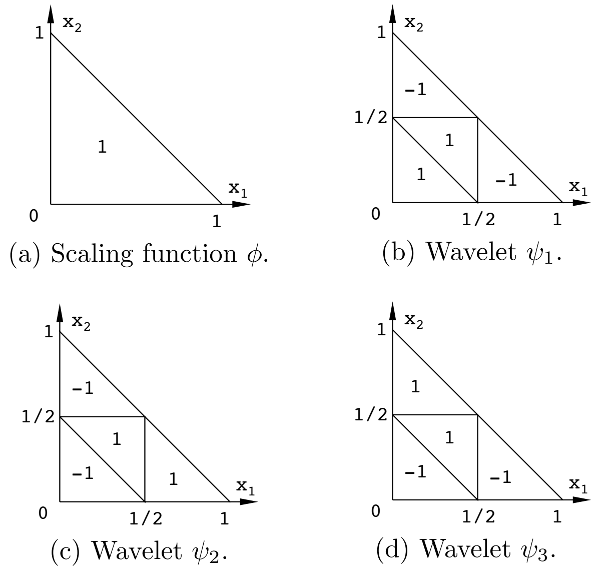
Fig. 7 shows the storage requirement of the matrix **A** derived through discretization with the Haar wavelets. The present example is the 2-D mixed boundary value problem illustrated in **Fig. 1**(b). In the present example, matrix compression based on Schneider's algorithm is restricted to the matrix **H**; the threshold determined by Eqs.(8) or (9) is not applicable to the matrix **G**. Nevertheless, Schneider's matrix compression does not cause to break the convergence of BE solution. The coefficient matrix compressed with Schneider's algorithm however, has more entries than that for Beylkin-type compression. The storage requirement for the Case 1, in particular, is remarkably large, as shown in **Fig. 7**. Little merit of Schneider's truncation based on Eq.(8) in matrix compression can thus be found in actual BE analysis for 2-D mixed boundary value problems. The numerical results depicted in **Fig. 7** include also the storage requirement of the matrix compressed by the *hybrid* algorithm with the above two truncation schemes. In the hybrid algorithm the Beylkin-type scheme is used to compress the matrix **G**, while the small entries of the matrix **H** are truncated on the basis of the threshold for Schneider's technique. The corresponding results, labelled "Hybrid", show better performance in matrix compression than that of the original Beylkin-type truncation.

(4) 3-D Neumann problem for external domain

We finally investigate the performance of the abovementioned truncation schemes for 3-D Laplace problem. In the present work, we deal with the external Neumann problem shown in **Fig. 2**. The potential u on boundary was approximated using the piecewise constant surface wavelet with a triangular support. This wavelet has the first-order vanishing moment; $p = 0$ and $n = 1$. The basis for the wavelet series then consists of the scaling function ϕ and three kind of wavelets ψ_1, ψ_2 and ψ_3. The DOF of this series increases at four times with the increase of the finest level m. **Fig. 8** illustrates the shape of four basis functions. In this figure, the number in every triangular subdomain indicates the height of the piecewise constant function defined in the corresponding patch.

Fig. 9 shows the storage requirement of the compressed coefficient matrix **A**, equivalent to the matrix **H**, for the 3-D example. The numerical results were obtained through discretization with $p = 0$ and $n = 1$. As shown in **Fig. 9**, the behaviour of storage requirement for 3-D analysis is similar to that for 2-D. In the DOF range where the finest level is set to a small value, the Beylkin-type technique enables us to compress the coefficient matrix at higher rates than Schneider's truncation scheme in Case 1 based on the appropriate threshold of Eq.(8). The superiority of the Beylkin-type scheme can also be found in BE analysis with about 200,000 DOF.

Schneider's algorithm for Case 2 was imple-



(a) Scaling function ϕ. (b) Wavelet ψ_1. (c) Wavelet ψ_2. (d) Wavelet ψ_3.

Fig.8 Piecewise constant surface wavelets with triangular supports. Three wavelets, ψ_1, ψ_2 and ψ_3, have the first-order vanishing moment. The number in every triangular patch indicates the height of the piecewise constant function in the corresponding patch.

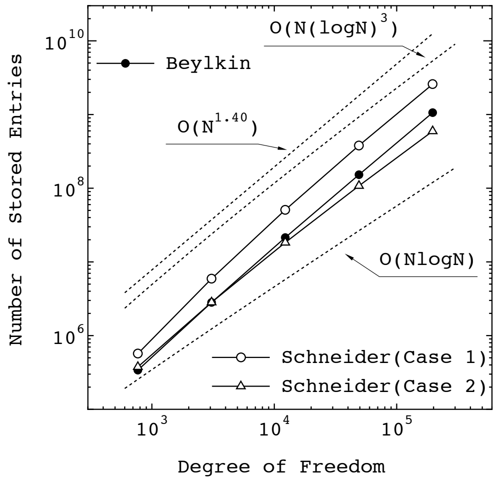


Fig.9 The number of stored entries in the coefficient matrix **A** for 3-D external Neumann problem illustrated in **Fig. 2**. The potential on boundary was approximated using the piecewise constant surface wavelets with the first-order vanishing moments and triangular support. The inappropriate threshold for the Case 2 was determined using Eq.(9) with $p' = 1$.

mented with an inappropriate threshold determined with $p' = 1$. Then, the asymptotic convergence rate of BE solution is not always retained due to overtruncation. The numerical results corresponding to Case 2 show the storage requirement comparable to that for the Beylkin-type compression without accuracy loss. However, the inappropriate threshold in Schneider's technique has to be used carefully so that the basis with the parameter p' can retain the rate of convergence.

In the present numerical test, the threshold for Beylkin-type matrix compression was determined with $\beta = 0.80$. The storage requirement of the coefficient matrix is thus predicted using the present estimation to $O(N^{1.40})$; the number of stored entries in actual BE analysis shows $O(N^{1.44})$.

6. CONCLUSIONS

In the present work, we have discussed the performance of the matrix compression schemes used for wavelet-based BE analysis. The two types of schemes, the Beylkin-type and Schneider's level-dependent algorithms, have been considered. The effectiveness of the Beylkin-type truncation technique, in particular, has been investigated through comparison of the storage requirement of the compressed coefficient matrix. We have described the theoretical estimation of the storage requirement. Using the proposed estimation, the number of stored entries in the coefficient matrix has been predicted to show increasing behaviour of $O(N^{1+\gamma})$ $(0 < \gamma < 1)$ for Beylkin-type compression. This fact is the basis that the Beylkin-type scheme is inferior in storage requirement to Schneider's truncation technique which ensures $O(N(\log N)^{\alpha})$ $(\alpha \leq 1)$. The storage size required in numerical tests is increased at the same order predicted by the present estimation. However, the matrix compression rates for Beylkin-type truncation is higher than or comparable to that for Schneider's matrix compression.

We have also discussed the restrictions of Schneider's algorithm in the application to matrix compression. The restrictions depends on the order of a boundary integral operator and the kind of wavelets. Schneider's scheme is not applicable to certain discretization conditions, because the appropriate threshold cannot be determined with the original scheme. We thus have attempted to implement Schneider's algorithm with an inappropriate threshold. Matrix compression using the inappropriate threshold does not always retain the convergence rate of BE solution. In numerical tests, the choice of sufficiently small threshold out of applicable range causes overtruncation. The error of the solution then usually increases. We, hence, have to take care to retain the convergence of BE solution, if we use an inappropriate threshold in Schneider's scheme.

REFERENCES

1) Dahmen, W., Prössdorf, S. and Schneider, R.: Wavelet approximation methods for pseudodifferential equations II: Matrix compression and fast solution. *Adv. Comput. Math.*, Vol.1, pp.259–335, 1993.
2) Schneider, R: Multiskalen- und Wavelet-Matrixkompression: Analysisbasierte Methoden zur effizienten Lösung großer vollbesetzter Gleichungssysteme. B.G.Teubner, Stuttgart, 1998.
3) von Petersdorff, T. and Schwab, C.: Wavelet approximations for first kind boundary integral equations on polygons. *Numer. Math.*, Vol.74, pp.479–519, 1996.
4) von Petersdorff, T., Schwab, C. and Schneider, R.: Multiwavelets for second-kind integral equations. *SIAM J. Numer. Anal.*, Vol.34, No.6, pp.2212–2227, 1997.
5) Lage, C. and Schwab, C.: Wavelet Galerkin algorithms for boundary integral equations. *SIAM J. Sci. Comput.*, Vol.20, No.6, pp.2195–2222, 1999.
6) Amaratunga, K. and Castrillon-Candas, J.E.: Surface wavelets: a multiresolution signal processing tool for 3D computational modelling. *Int. J. Numer. Meth. Engng.*, Vol.52, pp.239–271, 2001.
7) Rathsfeld, A.: A wavelet algorithm for the boundary element solution of a geodetic boundary value problem. *Comput. Meth. Appl. Mech. Engng.*, Vol.157, pp.267–287, 1998.
8) Goswami, J.C., Chan, A.K. and Chui, C.K.: On solving first-kind integral equations using wavelets on a bounded interval. *IEEE Trans. Antennas Propagat.*, Vol.43, No.6, pp.614–622, 1995.
9) Beylkin, G., Coifman, R. and Rokhlin, V.: Fast wavelet transforms and numerical algorithms I. *Comm. Pure Appl. Math.*, Vol.44, pp.144–183, 1991.
10) Wang, G.: A hybrid wavelet expansion and boundary element analysis of electromagnetic scattering form conducting objects. *IEEE Trans. Antennas Propagat.*, Vol.43, No.2, pp.170–178, 1995.
11) Koro, K. and Abe, K.: Non-orthogonal spline wavelets for boundary element analysis. *Engng. Anal. Bound. Elems.*, Vol.25, pp.149–164, 2001.
12) Koro, K. and Abe, K.: A practical determination strategy of optimal threshold parameter for matrix compression in wavelet BEM. *Int. J. Numer. Meth. Engng.*, Vol.57, pp.169–191, 2003.

(Received February 1, 2005)