THE FAST MULTIPOLE VIRTUAL BOUNDARY ELEMENT METHOD FOR SOLVING THIN PLATE PROBLEMS

Wei SI*

Department of Building Engineering, Tongji University, Shanghai 200092, China

ABSTRACT

The fast multipole method (FMM) has been used to reduce the computing operations and memory requirements in large numerical analysis problems through optimizing the computational structure. This paper presents a new virtual boundary element method (VBEM) for solving thin plate problems with the acceleration of FMM. In this thesis, the numerical scheme based on VBEM suitable for FMM with respect to thin plate problem of elasticity is derived, so that the amount of the computational elapsed time and the memory volume of the storage problems with the calculation of demand are linearly proportional to the number of degrees of freedom of the problem to be solved. The numerical results for the analysis of thin plate structures have proved the feasibility, efficiency and calculating precision of the method.

Keywords: fast multipole method (FMM) , virtual boundary element method (VBEM) , numerical scheme, thin plate.

1. INTRODUCTION

Since thin plates have a wide range of engineering application, some plate theories are derived to reduce the problem from three-dimension to two-dimension using corresponding basic assumptions and fundamental solutions. But the plate theory may not be satisfactory in some cases, especially in complex plate. In order to improve the calculation accuracy and expand the versatility, a three-dimensional virtual boundary element method which employs the Kelvin solution and the least square method is presented to deal with thin plate problems. Different basic assumptions and fundamental solutions are not used for solving different plate problems and the treatment of singular integration is rendered unnecessary.

BEM and VBEM requires $O(N^2)$ memory and $O(N^3)$ operations for a problem with N unknowns because the resulting system of equations has a dense coefficient matrix which hinders solutions to large-scale practical problems. Recently, given the fast multipole method (FMM) promising to reduce the CPU time and memory requirement to in multiplying a matrix by a vector (Jiang Yantao et al. 2008; L. Greengard and V. Rokhlin 1985; Hrycak T, Rokhlin V 1998; N. Nishimura 1999), so the introduction of the FMM to both VBEM and BEM may overcome the

* Corresponding author: Email: 2009swreal@tongji.edu.cn

limitation of their inability to solve the problems with large-scale DOF's on personal computers effectively and efficiently. This paper presents a new virtual boundary element method (VBEM) for Solving thin plate problems with the acceleration of FMM. The numerical scheme based on VBEM suitable for FMM with respect to thin plate problem of elasticity is derived, so that the memory requirement and computing operations for the solution of the dense matrix vector product can be reduced to $O(N)$. The result shows that the FM-VBEM is more efficient than the conventional VBEM in thin plate problem.

2. THE NUMERICAL SCHEME OF VIRTUAL BOUNDARY ELEMENT METHOD FOR SOLVING THIN PLATE PROBLEMS

The idea of virtual boundary element method for solving thin plate problems is put forward here and the numerical scheme can be derived as follow (Sun Huanchun et al. 1999; Xu Qiang and Sun Huanchun 2000).

The square variance functional $J_R\left[q(\xi)\right]$ about virtual source function $q(\xi)$ $(=\left\{q_1(\boldsymbol{\xi})\ \ q_2(\boldsymbol{\xi})\ \ q_3(\boldsymbol{\xi})\right\}^{\mathrm{T}})$ is established with employing Kelvin solution and the least square method. That is

$$\begin{aligned}J_R\left[q(\xi)\right]=&\int_{s'}\alpha\sum_{i=1}^{3}\left[p_i^1(\boldsymbol{x}')+p_i^2(\boldsymbol{x}'')-\tilde{p}_i(\boldsymbol{x})\right]^2\mathrm{d}S+\sum_{j=1}^{m_1+m_2}\alpha_1\left[w(\boldsymbol{x}_j)-\tilde{w}(\boldsymbol{x}_j)\right]^2\\&+\sum_{j=1}^{m_2}\alpha_2\left[\theta_i(\boldsymbol{x}_j)-\tilde{\theta}_i(\boldsymbol{x}_j)\right]^2+\sum_{j=1}^{m_1+m_3}\alpha_3\left[M_i(\boldsymbol{x}_j)-\tilde{M}_i(\boldsymbol{x}_j)\right]^2\\&+\sum_{j=1}^{m_3}\alpha_4\left[V_i(\boldsymbol{x}_j)-\tilde{V}_i(\boldsymbol{x}_j)\right]^2+\sum_{j=1}^{m_3}\alpha_5\left[Q_i(\boldsymbol{x}_j)-\tilde{Q}_i(\boldsymbol{x}_j)\right]^2\end{aligned}\tag{1}$$

where $\boldsymbol{x}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\boldsymbol{x}_3\}$ is a point on mid-surface and the upper boundary point and lower boundary point corresponding the mid-surface point $\boldsymbol{x}$ is $\boldsymbol{x}'=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\boldsymbol{x}_3\pm\frac{h}{2}\}$; $p_i^1(\boldsymbol{x}')$ and $p_i^2(\boldsymbol{x}'')$ are the traction value of the jth boundary point $\boldsymbol{x}_j$ along ith direction on the upper boundary point and lower boundary point, $\tilde{p}_i(\boldsymbol{x})$ are the sum of known traction value of the boundary point $\boldsymbol{x}_j$ along ith direction on the upper boundary point and the one on the lower boundary point; S' is area of mid-surface, $\boldsymbol{x}_j$ is boundary point on shell side; m_1 is the number of boundary nodes on simply-supported shell side, m_2 is the number of boundary nodes on fixed supported shell side, m_3 is the number of boundary nodes on free shell side; $w(\boldsymbol{x}_j)$ and $\tilde{w}(\boldsymbol{x}_j)$ are the calculated displacement value and the known displacement value of boundary point $\boldsymbol{x}_j$ on simply-supported shell side or fixed supported shell side; $\theta_i(\boldsymbol{x}_j)$ and $\tilde{\theta}_i(\boldsymbol{x}_j)$ are the calculated rotation angle value and the known rotation angle value of boundary point $\boldsymbol{x}_j$ on fixed supported shell side; $M_i(\boldsymbol{x}_j)$ and $\tilde{M}_i(\boldsymbol{x}_j)$ are the calculated bending moment value and the known bending moment value of boundary point $\boldsymbol{x}_j$ on simply-supported shell side or free shell side; $V_i(\boldsymbol{x}_j)$ and $\tilde{V}_i(\boldsymbol{x}_j)$ are the calculated transverse force value and the known transverse force value of boundary point $\boldsymbol{x}_j$ on free shell side; $Q_i(\boldsymbol{x}_j)$ and $\tilde{Q}_i(\boldsymbol{x}_j)$ are the calculated in-plane shear force value and the known in-plane shear force value of boundary point $\boldsymbol{x}_j$ on free shell side; α , $\alpha_1 \sim \alpha_5$ are weight coefficient about the corresponding physical quantities respectively.

$$p_i(\boldsymbol{x}) = \int_{s'} p_{ik}^*(\boldsymbol{x}', \boldsymbol{\xi}) q_k(\boldsymbol{\xi}) dS' + \int_{s'} p_{ik}^*(\boldsymbol{x}'', \boldsymbol{\xi}) q_k(\boldsymbol{\xi}) dS' \quad (i=1,2,3) \tag{2}$$

$$w(\boldsymbol{x}) = \int_{s'} u_{3k}^*(\boldsymbol{x}, \boldsymbol{\xi}) q_k(\boldsymbol{\xi}) dS' \tag{3}$$

$$\theta_i(\boldsymbol{x}) = \int_{s'} \theta_{ik}^*(\boldsymbol{x}, \boldsymbol{\xi}) q_k(\boldsymbol{\xi}) dS' \quad (i=1,2) \tag{4}$$

$$M_i(\boldsymbol{x}) = \int_{s'} \left[\int_{-h/2}^{h/2} \sigma_{iik}^*(\boldsymbol{x}, \boldsymbol{\xi}) x_3 dx_3 \right] q_k(\boldsymbol{\xi}) dS' \quad (i=1,2) \tag{5}$$

$$V_i(\boldsymbol{x}) = \int_{s'} \left[\int_{-h/2}^{h/2} \sigma_{i3k}^*(\boldsymbol{x}, \boldsymbol{\xi}) dx_3 \right] q_k(\boldsymbol{\xi}) dS' \quad (i=1,2) \tag{6}$$

$$Q_i(\boldsymbol{x}) = \int_{s'} \left[\int_{-h/2}^{h/2} \sigma_{ijk}^*(\boldsymbol{x}, \boldsymbol{\xi}) dx_3 \right] q_k(\boldsymbol{\xi}) dS' \quad (i=1, j=2;\ i=2, j=1) \tag{7}$$

where u_{ik}^*, θ_{ik}^* and σ_{ijk}^* are the Kelvin solution of displacement, rotation angle and stress to 3-D problem of elasticity.

In order to obtain the solution $q(\xi)$, the variation is being made from

$$\delta \boldsymbol{J} = 0 \tag{8}$$

Eq. (8) is often called least squares variation equation. Equation (8) can be rewritten in matrix form as follows

$$\boldsymbol{A}\varphi = \boldsymbol{B} \tag{9}$$

where $\boldsymbol{A}$ is the dense coefficient matrix. φ is the vector of unkown virtual load. Once the virtual function is obtained, the displacement and stress at any point can be calculated according to the principle of superposition and the fundamental solutions.

3. FAST MULTIPOLE VIRTUAL BOUNDARY ELEMENT METHOD FOR SOLVING THIN PLATE PROBLEMS

The main character of original FMM is to store the matrix–vector multiplications in the linear system by means of tree structure in every iteration, which can reduce the CPU time and memory requirement to $O(N)$ and is efficient in solving large-scale problems. Specific four steps of this algorithm are shown in fig 1 and described as follows: (1) Multipole expansion; (2) Moment-to-moment translation (M2M); (3) Local expansion and moment-to-local translation (M2L); (4) Local expansion to local expansion translation (L2L). The steps are illustrated by the case of angle rotation and its integral here.

3.1. Multipole expansion

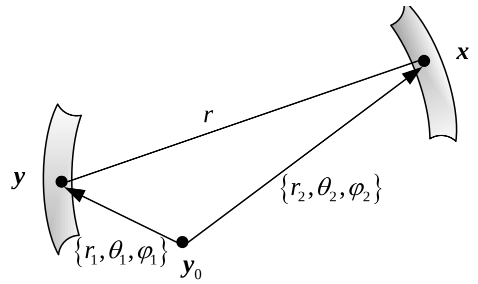


Fig.1 Requirements for spherical harmonic expansion

Assuming $\boldsymbol{y}_0$ to be a point close to the source point $\boldsymbol{y}$ (Fig. 1), that is, $\left|\overrightarrow{\boldsymbol{y}_0\boldsymbol{y}}\right| < \left|\overrightarrow{\boldsymbol{y}_0\boldsymbol{x}}\right|$, the integral in (4) can be derived as follows

$$\begin{aligned}\theta_j(\boldsymbol{x}) &= \int_{S'} \theta^*_{ji}(\boldsymbol{x},\boldsymbol{y}) q_i(\boldsymbol{y}) dS' \\ &= \frac{1}{8\pi\mu}\sum_{n=0}^{\infty}\sum_{m=-n}^{n}\left[\overline{E^S_{ij,n,m}}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{x}})\int_S R_{n,m}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{y}})\varphi_i(\boldsymbol{y})ds(\boldsymbol{y}) + \overline{H^S_{ij,n,m}}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{x}})\int_S (\overrightarrow{\boldsymbol{y}_0\boldsymbol{y}})_3 R_{n,m}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{y}})\varphi_i(\boldsymbol{y})ds(\boldsymbol{y})\right] \\ &= \frac{1}{8\pi\mu}\sum_{n=0}^{\infty}\sum_{m=-n}^{n}\left[\overline{E^S_{ij,n,m}}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{x}})\mathrm{M}^1_{i,n,m}(\boldsymbol{y}_0) + \overline{H^S_{ij,n,m}}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{x}})\mathrm{M}^2_{i,n,m}(\boldsymbol{y}_0)\right]\end{aligned} \tag{10}$$

$$E^S_{ij,n,m}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{x}}) = t_1\delta_{i3}\frac{\partial}{\partial x_j}S_{n,m}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{x}}) - t_2\frac{\partial}{\partial x_j}\left((\overrightarrow{\boldsymbol{y}_0\boldsymbol{x}})_3\frac{\partial}{\partial x_i}S_{n,m}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{x}})\right) \tag{11}$$

$$H^{S}_{ij,n,m}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{x}}) = t_2 \frac{\partial^2}{\partial x_i \partial x_j} S_{n,m}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{x}}) \tag{12}$$

$$t_1 = \frac{\lambda + 3\mu}{\lambda + 2\mu} \qquad t_2 = \frac{\lambda + \mu}{\lambda + 2\mu} \tag{13}$$

Where, the multipole expansion

$$\mathrm{M}^{1}_{i,n,m}(\boldsymbol{y}_0) = \int_S R_{n,m}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{y}})\varphi_i(\boldsymbol{y})ds(\boldsymbol{y}) \tag{14}$$

$$\mathrm{M}^{2}_{i,n,m}(\boldsymbol{y}_0) = \int_S (\overrightarrow{\boldsymbol{y}_0\boldsymbol{y}})_3 R_{n,m}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{y}})\varphi_i(\boldsymbol{y})ds(\boldsymbol{y}) \tag{15}$$

$\mathrm{M}^{1}_{i,n,m}(\boldsymbol{y}_0)$ and $\mathrm{M}^{2}_{i,n,m}(\boldsymbol{y}_0)$ are called moments about $\boldsymbol{y}_0$, which are independent of the field point $\boldsymbol{x}$ and only need to be computed once. Once these moments are obtained, the integral for any field point $\boldsymbol{x}$ can be evaluated using (10).

3.2. Moment-to-moment translation (M2M)

According the intrinsic property of spherical harmonic function (16),

$$R_{n,m}(\overrightarrow{\boldsymbol{y}_c\boldsymbol{y}}) = \sum_{n'=0}^{n} \sum_{m'=-n'}^{n'} R_{n-n',m-m'}(\overrightarrow{\boldsymbol{y}_0\boldsymbol{y}}) R_{n',m'}(\overrightarrow{\boldsymbol{y}_c\boldsymbol{y}_0}) \tag{16}$$

we obtain the new moments about $\boldsymbol{y}_c$ by old moment translation

$$\mathrm{M}^{1}_{i,n,m}(\boldsymbol{y}_c) = \sum_{n'=0}^{n} \sum_{m'=-n'}^{n'} R_{n',m'}(\overrightarrow{\boldsymbol{y}_c\boldsymbol{y}_0}) \mathrm{M}^{1}_{i,n-n',m-m'}(\boldsymbol{y}_0) \tag{17}$$

$$\mathrm{M}^{2}_{i,n,m}(\boldsymbol{y}_c) = \sum_{n'=0}^{n} \sum_{m'=-n'}^{n'} R_{n',m'}(\overrightarrow{\boldsymbol{y}_c\boldsymbol{y}_0}) \left[\mathrm{M}^{2}_{i,n-n',m-m'}(\boldsymbol{y}_0) - (\overrightarrow{\boldsymbol{y}_0\boldsymbol{y}_c})_3 \mathrm{M}^{1}_{i,n-n',m-m'}(\boldsymbol{y}_0) \right] \tag{18}$$

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

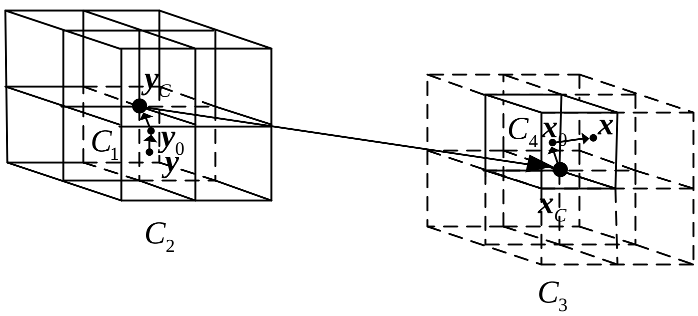


Fig.2 The schematic diagram of the expansion and translation

Suppose that $\boldsymbol{x}_c$ is a point close to the field point $\boldsymbol{x}$ (Fig. 2), that is, $\left|\overrightarrow{\boldsymbol{x}_c\boldsymbol{x}}\right| < \left|\overrightarrow{\boldsymbol{y}_c\boldsymbol{x}_c}\right|$, From the integral expansion (10) and using the intrinsic property of spherical harmonic function (19)

$$\overline{S_{n,m}}(\overrightarrow{\boldsymbol{y}_c\boldsymbol{x}}) = \sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'}\left(-1\right)^{n'} R_{n',m'}(\overrightarrow{\boldsymbol{x}_c\boldsymbol{x}})\overline{S_{n+n',m+m'}}(\overrightarrow{\boldsymbol{y}_c\boldsymbol{x}_c}) \tag{19}$$

we have

$$\int_S \theta_{ji}^*(\boldsymbol{x},\boldsymbol{y})\varphi_i(\boldsymbol{y})ds(\boldsymbol{y}) = \frac{1}{8\pi\mu}\sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'} E_{ij,n',m'}^R(\overrightarrow{\boldsymbol{x}_c\boldsymbol{x}})L_{i,n',m'}^1(\boldsymbol{x}_c) + \frac{1}{8\pi\mu}\sum_{n=0}^{\infty}\sum_{m=-n}^{n} H_{ij,n,m}^R(\overrightarrow{\boldsymbol{x}_c\boldsymbol{x}})L_{i,n',m'}^2(\boldsymbol{x}_c) \tag{20}$$

where the coefficients $\mathrm{L}_{i,n,m}^1(\boldsymbol{x}_c)$ $\mathrm{L}_{i,n,m}^2(\boldsymbol{x}_c)$ are given by the following translation

$$\mathrm{L}_{i,n,m}^1(\boldsymbol{x}_c) = \sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'}\left(-1\right)^{n}\overline{S_{n+n',m+m'}}(\overrightarrow{\boldsymbol{y}_c\boldsymbol{x}_c})\mathrm{M}_{i,n',m'}^1(\boldsymbol{y}_c) \tag{21}$$

$$\mathrm{L}_{i,n,m}^2(\boldsymbol{x}_c) = \sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'}\left(-1\right)^{n}\overline{S_{n+n',m+m'}}(\overrightarrow{\boldsymbol{y}_c\boldsymbol{x}_c})\left(\mathrm{M}_{i,n',m'}^2(\boldsymbol{y}_c) - (\overrightarrow{\boldsymbol{y}_c\boldsymbol{x}_c})_3\mathrm{M}_{i,n',m'}^1(\boldsymbol{y}_c)\right) \tag{22}$$

these are called multipole moments to local expansion translation (M2L).

3.4. Local expansion to local expansion translation (L2L)

By the intrinsic property of spherical harmonic function, the integral expansion (10) can be expressed as

$$\int_S \theta_{ji}^*(\boldsymbol{x},\boldsymbol{y})\varphi_i(\boldsymbol{y})ds(\boldsymbol{y}) = \frac{1}{8\pi\mu}\sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'} E_{ij,n',m'}^R(\overrightarrow{\boldsymbol{x}_0\boldsymbol{x}})\mathrm{L}_{i,n',m'}^1(\boldsymbol{x}_0) + \frac{1}{8\pi\mu}\sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'} H_{ij,n',m'}^R(\overrightarrow{\boldsymbol{x}_0\boldsymbol{x}})\mathrm{L}_{i,n',m'}^2(\boldsymbol{x}_0) \tag{23}$$

Where the new coefficients $\mathrm{L}_{i,n,m}^1(\boldsymbol{x}_0)$ and $\mathrm{L}_{i,n,m}^2(\boldsymbol{x}_0)$ are given by the following L2L as follows

$$\mathrm{L}_{i,n,m}^1(\boldsymbol{x}_0) = \sum_{n'=n}^{\infty}\sum_{m'=-n'}^{n'} R_{n'-n,m'-m}(\overrightarrow{\boldsymbol{x}_c\boldsymbol{x}_0})\mathrm{L}_{i,n',m'}^1(\boldsymbol{x}_c) \tag{24}$$

$$\mathrm{L}_{i,n,m}^2(\boldsymbol{x}_0) = \sum_{n'=n}^{\infty}\sum_{m'=-n'}^{n'} R_{n'-n,m'-m}(\overrightarrow{\boldsymbol{x}_c\boldsymbol{x}_0})\left(\mathrm{L}_{i,n',m'}^2(\boldsymbol{x}_c) - (\overrightarrow{\boldsymbol{x}_c\boldsymbol{x}_0})_3\mathrm{L}_{i,n',m'}^1(\boldsymbol{x}_c)\right) \tag{25}$$

This translation between local moments is applied when the local-expansion center shifts, which are called local expansion to local expansion translation (L2L).

3.5. Numerical Implementation

The first step of the fast multipole algorithm is the construction of a Octree structure. The second step is a iteration process, which includes computation of the multipole moments by Upward and the local moments by Downward. Upward and Downward computations are executed iteratively until the defined accuracy is achieved, and the generalized minimal residual (GMRES) method is used as the iterative solver in this analysis.

4. NUMERICAL EXAMPLES

Consider a $20 \times 210 \times 210 mm$ thin plate with fixed supported shell sides as shown in Fig.3. A transverse uniform pressure $q = 20$ KN is applied to the top surface. The plate material properties are Young modulus $E = 200$ GPa and Poisson ratio $\nu = 0.25$. This example is studied here in order to validate the efficiency and precision of FMM-VBEM for thin plate problem proposed.

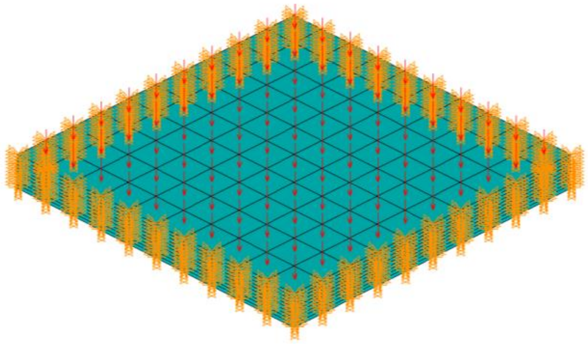
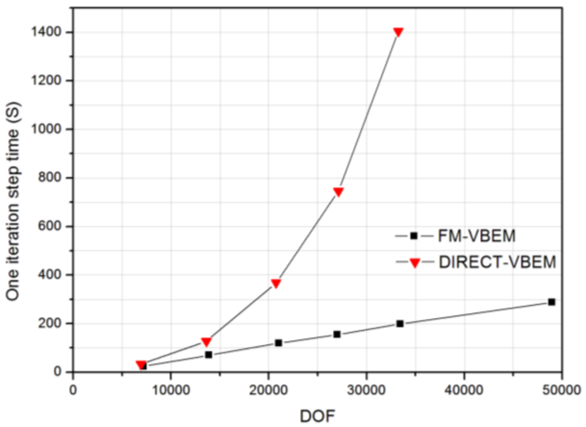


Fig. 3 Thin plate with fixed supported shell sides



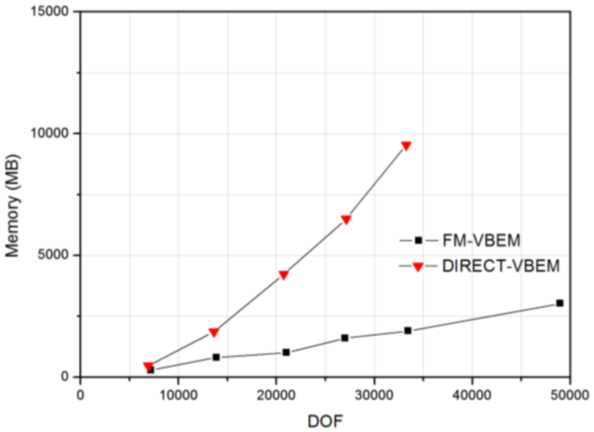


Fig. 4 One iteration step time comparison **Fig. 5 Memory requirements comparison**

Figure 4 gives one iteration step time of both direct VBEM and the FM-BEM for different DOF. The FM-BEM is considerably faster than the direct VBEM when DOF exceeds about 20000. The FM-BEM shows a linear dependence of one iteration step time on the DOF, while direct VBEM is an $O(N^3)$.

Figure 5 indicates the memory requirements comparison between direct VBEM and FM-VBEM. The memory requirements for the FM-BEM are $O(N)$, but are $O(N^2)$ for the conventional VBEM.

Now take the z-axial computational displacement of central point on the thin plate by the FM-VBEM and the direct VBEM to verify the precision. When DOF= 33060, FM-VBEM and the direct VBEM have the result 0.0126493 qa^4 / D and 0.01264945 qa^4 / D individually. Here the analytical solution of central point is 0.01265 qa^4 / D . The results indicate both original FMM-VBEM and direct VBEM have almost same precision and both are very close to analytical solution.

Illustrated by example, the algorithm presented in this paper has achieved the expected feasibility, efficiency and calculating precision.

5. CONCLUSIONS

The fast multipole virtual boundary element method for solving thin plate problems significantly improve the computational efficiency and reduce the memory requirement. Numerical examples show that the technique is more efficient than the conventional VBEM in thin plate problem, especially for the large-scale ones.

REFERENCES

Sun Huanchun, Zhang Lizhou, Xu Qiang, et al (1999). Non-singularity boundary element methods. Dalian Universit y of Technology Press.

Xu Qiang, Sun Huanchun (2000). Three-dimensional virtual boundary element analysis of plate bending. [J]. Engineering Mechanics. 17(3), 23-30.

Jiang Yantao, Xu Qiang, and Zhang Zhijia (2008). Fast Multipole VBEM for Analysis of Tall Shear wall Structures. Structural Engineers. 24(4).

L. Greengard, V. Rokhlin (1985). A Fast Algorithm for Particle Simulations. Comput. Phys. 60: 187-207.

Hrycak T, Rokhlin V (1998). An improved fast multipole algorithm for potential fields. SIA M J Sci Comput. 19: l804-1826.

N. Nishimura, K. Yoshida, and S. Kobayashi (1999). A fast multi-pole boundary integral equation method for crack problems in 3D, Eng Anal Bound Elem. 23, 97-105.