Mechanical Analysis of 3D Composite Materials by Hybrid Boundary Node Method

Yu Miao[1], Zhe Chen[1], Qiao Wang[1,2] and Hongping Zhu[1]

Abstract: In this paper, an improved multi-domain model based on the hybrid boundary node method (Hybrid BNM) is proposed for mechanical analysis of 3D composites. The Hybrid BNM is a boundary type meshless method which based on the modified variational principle and the Moving Least Squares (MLS) approximation. The improved multi-domain model can reduce the total degrees of freedom (DOFs) compared with the conventional multi-domain solver. It is very suitable for the inclusion-based composites, especially for the composites when the inclusions are solid and totally embedded in the matrix domain. Numerical examples are presented to verify the improved multi-domain model and the results have shown the accuracy and efficiency of the improved model.

Keywords: Hybrid boundary node method, mechanical analysis, Composites, Improved multi-domain model.

1   Introduction

Composite materials become more and more important in industrial projects. There has been much interest in the modeling and simulations of composites in order to characterize their mechanical properties for engineering applications. Much effort, time and expense would be saved if the properties of composites could be predicted accurately. Many numerical models based on finite element method (FEM) and boundary element method (BEM) have been developed to simulate composite materials so far. Hou and Wu (1997) studied a multiscale finite element method for solving a class of elliptic problems arising from composite materials and flows in porous media. A numerical homogenization technique based on the FEM with representative volume element (RVE) was used to evaluate the effective material

---

[1] School of Civil Engineering and Mechanics, Huazhong University of Science and Technology, Wuhan 430074, China.

[2] Corresponding Author.
   Tel: 86 27 87540172; Fax: 86 27 87542231; E-mail: hust.qiaowang@gmail.com

properties with periodic boundary conditions by Kari, Berger, Rodriguez-Ramos and Gabbert (2007). Segurado and LLorca (2004) developed a new 3D quadratic interface finite element to simulate fracture in composite materials Hybrid/Mixed finite elements and mixed methods [Bishay and Atluri (2012); Bishay, Sladek, Sladek and Atluri (2012); Dong, Alotaibi, Mohiuddine and Atluri (2014); Dong, El-Gizawy, Juhany and Atluri (2014)] were developed recently and applied for mechanical modeling of composites. Eischen and Torquato (1993) applied the BEM to determine the effective elastic moduli of continuum models of composite materials. An advanced 3D boundary element method was proposed by Chen and Liu (2005) for characterizations of composite materials. Liu, Nishimura, Otani, Takahashi, Chen and Munakata (2005) developed a new BEM for 3D analysis of fiber-reinforced composites based on a rigid-inclusion model. A new BEM for the numerical analysis of mechanical properties in 3D particle-reinforced composites was proposed by Wang and Yao (2005). Yao, Xu, Wang and Zheng (2009) applied the BEM on the simulation of carbon nanotube (CNT) composites, including the simulation of elastic, thermal and electric properties. Since the BEM has a dense matrix in the final system equation, some researchers [Nishimura and Liu (2004); Liu, Nishimura and Otani (2005); Liu, Nishimura, Otani, Takahashi, Chen and Munakata (2005); Wang and Yao (2005); Wang and Yao (2008); Yao, Xu, Wang and Zheng (2009)] coupled the BEM with fast multipole method (FMM) [Rokhlin (1985)] in order to simulate composites with large scale computation.

Although the FEM and BEM are widely investigated and have been applied in many areas, they may not be convenient and efficient to simulate composites or fracture problems [Dong and Atluri (2013a); Dong and Atluri (2013b); Dong and Atluri (2013c)] because of the cells caused by the mesh procedure. In order to overcome the difficulty caused by meshing, meshless or meshfree methods are widely investigated in recent years. Many meshless methods have been proposed so far, including the smoothed particle hydrodynamics (SPH) method [Gingold and Monaghan (1977); Lucy (1977)], the reproducing kernel particle methods (RKPM) [Liu, Jun and Zhang (1995)], the hp-clouds method [Liszka, Duarte and Tworzydlo (1996)], the element free Galerkin method (EFG) [Belytschko, Lu and Gu (1994)], the meshless local Petrov-Galerkin (MLPG) approach [Atluri and Zhu (1998)], the boundary node method (BNM) [Mukherjee and Mukherjee (1997)], the Galerkin boundary node method (GBNM) [Li and Zhu (2009)], the boundary face method (BFM) [Zhang, Qin, Han and Li (2009)], the Method of Finite Sphere [De, Hong and Bathe (2003)], the boundary point interpolation method [Gu and Liu (2002)] and the hybrid boundary node method (Hybrid BNM) [Zhang and Yao (2001); Zhang, Yao and Li (2002)].

The Hybrid BNM [Zhang, Tanaka and Matsumoto (2004a); Zhang and Yao (2004)]

is based on the moving least squares (MLS) approximation and the hybrid displacement variational formula. It not only reduces the spatial dimensions by one like boundary element method (BEM) or BNM, but also does not require any cells neither for interpolation of the solution variables nor for the boundary integration. In fact, the Hybrid BNM requires only discrete nodes located on the surface of domain and its parametric representation. The Hybrid BNM has been developed by Miao et al [Miao, Wang and Yu (2005); Miao and Wang (2006)] and applied to some other areas, such as elastodynamics problems [Miao, Wang, Liao and Zheng (2009)], Helmholtz problems [Miao, Wang and Wang (2009)] and multi-domain problems [Wang, Zheng, Miao and Lv (2011)].

The meshless methods are very suitable for simulation of composite materials and some literatures have reported relative work. Zhang et al. [Zhang, Tanaka and Matsumoto (2004b); Zhang and Tanaka (2008)] simulated the CNT based composites by Hybrid BNM based on a simplified model. Singh, Tanaka and Endo (2007) applied the EFG in the thermal analysis of CNT based composites. Wang, Miao and Zhu (2013a) used a fast multipole Hybrid BNM (FM-HBNM) to simulate the composites in 3D elasticity. There are mainly two models while simulating composites by Hybrid BNM so far. One is the multi-domain solver [Wang, Zheng, Miao and Lv (2011); Wang, Miao and Zhu (2013a)], which is based on the continuity and equilibrium conditions across the interface. The other one is the simplified approach [Zhang, Tanaka and Matsumoto (2004b)], which treats the CNTs as heat superconductors. The multi-domain solver can be applied to general composites since no assumption is added and the unknown vectors for each material are assembled in the final system equation. The simplified approach is very suitable for the case when the inclusions are heat superconductors or rigid bodies and it can reduce the total degrees of freedom (DOFs) substantially, however, it is not suitable to simulate general composites as multi-domain solver.

In this paper, an improved multi-domain model for 3D composites with inclusions in elasticity problems base on Hybrid BNM is proposed. It has already been applied to thermal analysis of composites [Wang, Miao and Zhu (2013b)] successfully and we extend it to elasticity problems in this paper. In the improved model, no other assumption is added and it can be applied to general case like the multi-domain solver. Much important, it can reduce the total DOFs in the final system equation as the simplified approach, and we will find out only the matrix domain is needed to be modeled in some special case.

This paper is organized as follows. The hybrid BNM is introduced in the second section and the multi-domain solver for 3D elasticity problems is reviewed in the third section. The formulation for the improved multi-domain model is derived in the fourth section. Then numerical examples are given in the fifth section.

2 The single-domain Hybrid BNM for 3D elasticity problems

In this section, the Hybrid BNM for 3D elasticity problems [Miao and Wang (2006)] is reviewed. The Hybrid BNM is based on a modified variational principle. In 3D elasticity problems, functions in the modified variational principle that assumed to be independent are: displacements $\tilde{u}_l$ and tractions $\tilde{t}_l$ on the boundary and displacements u_l inside the domain. Consider a domain Ω enclosed by $\Gamma = \Gamma_u + \Gamma_t$ with $\bar{u}_l$ and $\bar{t}_l$ are the prescribed displacements and tractions, respectively.

The displacements $\tilde{\mathbf{u}}$ and tractions $\tilde{\mathbf{t}}$ at the boundary Γ are approximated by the MLS approximation [Zhang and Yao (2004); Miao and Wang (2006)]

$$\tilde{\mathbf{u}}(\mathbf{s}) = \sum_{J=1}^{N} \Phi_J(\mathbf{s})\hat{\mathbf{u}}_J \tag{1}$$

$$\tilde{\mathbf{t}}(\mathbf{s}) = \sum_{J=1}^{N} \Phi_J(\mathbf{s})\hat{\mathbf{t}}_J \tag{2}$$

where N is the number of nodes located on the surface $\hat{\mathbf{u}}_J$ and $\hat{\mathbf{t}}_J$ are nodal values, and $\Phi_J(\mathbf{s})$ is the shape function of the MLS approximation, corresponding to node $\mathbf{s}_J$.

The $\mathbf{u}$ and $\mathbf{t}$ inside the domain can be approximated by fundamental solutions as

$$\mathbf{u} = \left\{ \begin{array}{c} u_1 \\ u_2 \\ u_3 \end{array} \right\} = \sum_{J=1}^{N} \left[\begin{array}{ccc} u_{11}^J & u_{21}^J & u_{31}^J \\ u_{12}^J & u_{22}^J & u_{32}^J \\ u_{13}^J & u_{23}^J & u_{33}^J \end{array} \right] \left\{ \begin{array}{c} x_1^J \\ x_2^J \\ x_3^J \end{array} \right\} \tag{3}$$

$$\mathbf{t} = \left\{ \begin{array}{c} t_1 \\ t_2 \\ t_3 \end{array} \right\} = \sum_{J=1}^{N} \left[\begin{array}{ccc} t_{11}^J & t_{21}^J & t_{31}^J \\ t_{12}^J & t_{22}^J & t_{32}^J \\ t_{13}^J & t_{23}^J & t_{33}^J \end{array} \right] \left\{ \begin{array}{c} x_1^J \\ x_2^J \\ x_3^J \end{array} \right\} \tag{4}$$

where $u_{lj}^J = u_{lj}(\mathbf{s}_J, Q)$ and $t_{lj}^J = t_{lj}(\mathbf{s}_J, Q)$ are the fundamental solutions x_l^J are unknown parameters. For 3D elasticity problems, the fundamental solutions are

$$u_{lj}^J = \frac{-1}{16\pi r(1-v)\mu}\{(3-4v)\delta_{lj} - r_{,l}r_{,j}\} \tag{5}$$

$$t_{lj}^J = \frac{-1}{8\pi(1-v)r^2}\{[(1-2v)\delta_{lj} + 3r_{,l}r_{,j}]\frac{\partial r}{\partial n} + (1-2v)(r_{,l}n_j - r_{,j}n_l)\} \tag{6}$$

where $r = r(\mathbf{s}_J, Q)$ and Q is the field point while $\mathbf{s}_J$ is the source point.

As the modified variational principle holds both in the whole domain and any sub-domain, the local sub-domain around each node can be taken into consideration.

Then the following set of equations, expressed in matrix form, are given as [Zhang and Yao (2004); Miao and Wang (2006)]

$$\mathbf{Ux} = \mathbf{H\hat{u}} \tag{7}$$

$$\mathbf{Tx} = \mathbf{H\hat{t}} \tag{8}$$

where

$$U_{IJ} = \int_{\Gamma_I} \begin{bmatrix} u^J_{11} & u^J_{21} & u^J_{31} \\ u^J_{12} & u^J_{22} & u^J_{32} \\ u^J_{13} & u^J_{23} & u^J_{33} \end{bmatrix} h_I(Q)\mathrm{d}\Gamma \tag{9}$$

$$T_{IJ} = \int_{\Gamma_I} \begin{bmatrix} t^J_{11} & t^J_{21} & t^J_{31} \\ t^J_{12} & t^J_{22} & t^J_{32} \\ t^J_{13} & t^J_{23} & t^J_{33} \end{bmatrix} h_I(Q)\mathrm{d}\Gamma \tag{10}$$

$$H_{IJ} = \int_{\Gamma_I} \begin{bmatrix} \Phi_J(\mathbf{s}) & 0 & 0 \\ 0 & \Phi_J(\mathbf{s}) & 0 \\ 0 & 0 & \Phi_J(\mathbf{s}) \end{bmatrix} h_I(Q)\mathrm{d}\Gamma \tag{11}$$

$$\mathbf{x^T} = [x^1_1, x^1_2, x^1_3, \cdots, x^N_1, x^N_2, x^N_3] \tag{12}$$

$$\mathbf{\hat{u}^T} = [\hat{u}^1_1, \hat{u}^1_2, \hat{u}^1_3, \cdots, \hat{u}^N_1, \hat{u}^N_2, \hat{u}^N_3] \tag{13}$$

$$\mathbf{\hat{t}^T} = [\hat{t}^1_1, \hat{t}^1_2, \hat{t}^1_3, \cdots, \hat{t}^N_1, \hat{t}^N_2, \hat{t}^N_3] \tag{14}$$

where $h_I(Q)$ is a weight function, Γ_I is a regularly shaped local region around node $\mathbf{s}_I$ in the parametric representation space of the boundary surface. Therefore, the integrals in Equations (9), (10) and (11) can be computed without using boundary elements.

For a general problem, either $\tilde{u}_l$ or $\tilde{t}_l$ can be known at each node on the boundary and by rearranging Equations (7) and (8), a final algebraic equation in terms of $\mathbf{x}$ only can be obtained as below:

$$\mathbf{Ax} = \mathbf{d} \tag{15}$$

For the node $\mathbf{s}_I$, if $\tilde{u}_l$ is known, select the correspond row in $\mathbf{U}$ to $\mathbf{A}$, otherwise, select the correspond row in $\mathbf{T}$ to $\mathbf{A}$, and the corresponding term of $\mathbf{d}$ comes from the matrix-vector product of $\mathbf{H\hat{u}}$ or $\mathbf{H\hat{t}}$. Then the unknown vector $\mathbf{x}$ is obtained by solving the final algebraic equation. The nodal values $\mathbf{\hat{u}}$ and $\mathbf{\hat{t}}$ on the boundary can be computed by the back-substitution of $\mathbf{x}$ into Equations (7) and (8)

3   The multi-domain Hybrid BNM for 3D elasticity problems

In this section, the multi-domain Hybrid BNM for 3D elasticity problems [Wang, Zheng, Miao and Lv (2011)] is introduced. Consider a problem with n sub-domains and N_i nodes are distributed on the bounding surface of subdomain-i. Then for each sub-domain, we can have the following equations from Equations (7) and (8)

$$\mathbf{U}^i\mathbf{x}^i = \mathbf{H}^i\hat{\mathbf{u}}^i \tag{16}$$

$$\mathbf{T}^i\mathbf{x}^i = \mathbf{H}^i\hat{\mathbf{t}}^i \tag{17}$$

where superscript i indicates the subdomain i.

The boundary nodes of subdomain-i can be sorted into n groups according to the locations of the nodes. Group i contains nodes that belong exclusively in subdomain-i and group j contains nodes that are on the interface with subdomain-j, where j is from 1 to n and $j \neq i$. Accordingly, Equations (16) and (17) are partitioned into blocked matrix equations as

$$
\begin{bmatrix}
\mathbf{U}^i_{11} & \mathbf{U}^i_{12} & \cdots & \mathbf{U}^i_{1n} \\
\mathbf{U}^i_{21} & \mathbf{U}^i_{22} & \cdots & \mathbf{U}^i_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{U}^i_{n1} & \mathbf{U}^i_{n2} & \cdots & \mathbf{U}^i_{nn}
\end{bmatrix}
\begin{Bmatrix}
\mathbf{x}^i_1 \\
\mathbf{x}^i_2 \\
\vdots \\
\mathbf{x}^i_n
\end{Bmatrix}
=
\begin{Bmatrix}
\mathbf{H}^i_1\hat{\mathbf{u}}^i_1 \\
\mathbf{H}^i_2\hat{\mathbf{u}}^i_2 \\
\vdots \\
\mathbf{H}^i_n\hat{\mathbf{u}}^i_n
\end{Bmatrix}
\tag{18}
$$

$$
\begin{bmatrix}
\mathbf{T}^i_{11} & \mathbf{T}^i_{12} & \cdots & \mathbf{T}^i_{1n} \\
\mathbf{T}^i_{21} & \mathbf{T}^i_{22} & \cdots & \mathbf{T}^i_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{T}^i_{n1} & \mathbf{T}^i_{n2} & \cdots & \mathbf{T}^i_{nn}
\end{bmatrix}
\begin{Bmatrix}
\mathbf{x}^i_1 \\
\mathbf{x}^i_2 \\
\vdots \\
\mathbf{x}^i_n
\end{Bmatrix}
=
\begin{Bmatrix}
\mathbf{H}^i_1\hat{\mathbf{t}}^i_1 \\
\mathbf{H}^i_2\hat{\mathbf{t}}^i_2 \\
\vdots \\
\mathbf{H}^i_n\hat{\mathbf{t}}^i_n
\end{Bmatrix}
\tag{19}
$$

where the superscript i denotes the subdomain-i; the subscript $k=1, 2,\ldots n$ denotes that the prescribed quantities are associated with the nodes in group k. The double subscripts ij, i and $j = 1,2,...n$ is used to convey the $\mathbf{U}$ $\mathbf{T}$ in Equations (16) and (17), by which the prescribed coefficient matrix blocks are computed, belong to group i and j, respectively.

At the interfaces between sub-domain i and j, the following continuity conditions exist. For the displacements

$$\left\{\mathbf{u}^j_i\right\} = \left\{\mathbf{u}^i_j\right\} \tag{20}$$

and for the tractions

$$\left\{\mathbf{t}^j_i\right\} = -\left\{\mathbf{t}^i_j\right\} \tag{21}$$

Consider a problem with two sub-domains, then for subdomain-1, we can have

$$\begin{bmatrix} \mathbf{U}_{11}^1 & \mathbf{U}_{12}^1 \\ \mathbf{U}_{21}^1 & \mathbf{U}_{22}^1 \end{bmatrix} \left\{ \begin{array}{c} \mathbf{x}_1^1 \\ \mathbf{x}_2^1 \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{H}_1^1 \hat{\mathbf{u}}_1^1 \\ \mathbf{H}_2^1 \hat{\mathbf{u}}_2^1 \end{array} \right\} \tag{22}$$

$$\begin{bmatrix} \mathbf{T}_{11}^1 & \mathbf{T}_{12}^1 \\ \mathbf{T}_{21}^1 & \mathbf{T}_{22}^1 \end{bmatrix} \left\{ \begin{array}{c} \mathbf{x}_1^1 \\ \mathbf{x}_2^1 \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{H}_1^1 \hat{\mathbf{t}}_1^1 \\ \mathbf{H}_2^1 \hat{\mathbf{t}}_2^1 \end{array} \right\} \tag{23}$$

For subdomain-2, we can have

$$\begin{bmatrix} \mathbf{U}_{11}^2 & \mathbf{U}_{12}^2 \\ \mathbf{U}_{21}^2 & \mathbf{U}_{22}^2 \end{bmatrix} \left\{ \begin{array}{c} \mathbf{x}_1^2 \\ \mathbf{x}_2^2 \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{H}_1^2 \hat{\mathbf{u}}_1^2 \\ \mathbf{H}_2^2 \hat{\mathbf{u}}_2^2 \end{array} \right\} \tag{24}$$

$$\begin{bmatrix} \mathbf{T}_{11}^2 & \mathbf{T}_{12}^2 \\ \mathbf{T}_{21}^2 & \mathbf{T}_{22}^2 \end{bmatrix} \left\{ \begin{array}{c} \mathbf{x}_1^2 \\ \mathbf{x}_2^2 \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{H}_1^2 \hat{\mathbf{t}}_1^2 \\ \mathbf{H}_2^2 \hat{\mathbf{t}}_2^2 \end{array} \right\} \tag{25}$$

Using Equations (20) and (21), Equations (22), (23), (24) and (25) can be assembled to an overall matrix equation as

$$\begin{bmatrix} \mathbf{A}_{11}^1 & \mathbf{A}_{12}^1 & 0 & 0 \\ \mathbf{U}_{21}^1 & \mathbf{U}_{22}^1 & -\mathbf{U}_{11}^2 & -\mathbf{U}_{12}^2 \\ \mathbf{T}_{21}^1 & \mathbf{T}_{22}^1 & \mathbf{T}_{11}^2 & \mathbf{T}_{12}^2 \\ 0 & 0 & \mathbf{A}_{21}^2 & \mathbf{A}_{22}^2 \end{bmatrix} \left\{ \begin{array}{c} \mathbf{x}_1^1 \\ \mathbf{x}_2^1 \\ \mathbf{x}_1^2 \\ \mathbf{x}_2^2 \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{d}_1^1 \\ 0 \\ 0 \\ \mathbf{d}_2^2 \end{array} \right\} \tag{26}$$

where $\mathbf{A}_{ij}^i$ and $\mathbf{d}_i^i$ are formed by merging $\mathbf{U}_{ij}^i$ and $\mathbf{T}_{ij}^i$, $\mathbf{H}_i^i \hat{\mathbf{u}}_i^i$ and $\mathbf{H}_i^i \hat{\mathbf{t}}_i^i$, respectively, according to the known boundary conditions.

If use $\mathbf{a}_{ij}$ stand for $\mathbf{U}_{jk}^i$ and $\mathbf{T}_{jk}^i$, $\mathbf{x}_i$ stand for $\mathbf{x}_j^i$, respectively, and ignore the minus symbol Equation (26) can be rewritten as below

$$\begin{bmatrix} \mathbf{a}_{11} & \\ \mathbf{a}_{12} & \mathbf{a}_{21} \\ & \mathbf{a}_{22} \end{bmatrix} \left\{ \begin{array}{c} \mathbf{x}_1 \\ \mathbf{x}_2 \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{d}_1^1 \\ 0 \\ \mathbf{d}_2^2 \end{array} \right\} \tag{27}$$

Similarly, for three subdomains, the final equation for the entire domain can be written as

$$\begin{bmatrix} \mathbf{a}_{11} & & \\ \mathbf{a}_{12} & \mathbf{a}_{21} & \\ & \mathbf{a}_{22} & \\ \mathbf{a}_{13} & & \mathbf{a}_{31} \\ & \mathbf{a}_{23} & \mathbf{a}_{32} \\ & & \mathbf{a}_{33} \end{bmatrix} \left\{ \begin{array}{c} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{d}_1^1 \\ 0 \\ \mathbf{d}_2^2 \\ 0 \\ 0 \\ \mathbf{d}_3^3 \end{array} \right\} \tag{28}$$

Equation (28) can be rewritten as

$$
\begin{bmatrix}
\mathbf{A}_2 & \\
\mathbf{a}_{13} & \mathbf{a}_{31} \\
& \mathbf{a}_{23} & \mathbf{a}_{32} \\
& & \mathbf{a}_{33}
\end{bmatrix}
\left\{
\begin{array}{c}
\mathbf{x}_1 \\
\mathbf{x}_2 \\
\mathbf{x}_3
\end{array}
\right\}
=
\left\{
\begin{array}{c}
\mathbf{D}_2 \\
\\
\mathbf{d}_3^3
\end{array}
\right\}
\tag{29}
$$

where $\mathbf{A}_2$ stands for the coefficient matrix of the two subdomains in the left hand side of Equation (27) and $\mathbf{D}_2$ stands for the vector in the right hand side of Equation (27). Then for n subdomains, the final equation for the entire domain can be written as

$$
\begin{bmatrix}
\mathbf{A}_{n-1} & \\
\mathbf{a}_{1n} & \mathbf{a}_{n1} \\
\ddots & \vdots \\
& \mathbf{a}_{nn}
\end{bmatrix}
\left\{
\begin{array}{c}
\mathbf{x}_1 \\
\mathbf{x}_2 \\
\vdots \\
\mathbf{x}_n
\end{array}
\right\}
=
\left\{
\begin{array}{c}
\mathbf{D}_{n-1} \\
\\
\mathbf{d}_n^n
\end{array}
\right\}
\tag{30}
$$

where $\mathbf{A}_{n-1}$ and $\mathbf{D}_{n-1}$ stand for the coefficient matrix and right hand vector of the n-1 subdomains.

The unknown vector $\mathbf{x}$ can be solved by the final matrix equation, and then by back-substitution $\mathbf{x}$ to matrix equations for each sub-domain, the boundary unknowns can be obtained either on the interfaces or the external boundary surfaces.

4   Improved multi-domain Hybrid BNM for 3D elasticity problems

For composite materials, the multi-domain solver introduced in section 3 is a natural way to be chosen. Now consider a matrix with inclusions as showed in Figure 1(a). S_0 is the sub-domain of matrix and $S_1, S_2, ..., S_n$ are sub-domains of the inclusions. For the matrix, Equations (18) and (19) can be rewritten as

$$
\begin{bmatrix}
\mathbf{U}_{00}^0 & \mathbf{U}_{01}^0 & \cdots & \mathbf{U}_{0n}^0 \\
\mathbf{U}_{10}^0 & \mathbf{U}_{11}^0 & \cdots & \mathbf{U}_{1n}^0 \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{U}_{n0}^0 & \mathbf{U}_{n1}^0 & \cdots & \mathbf{U}_{nn}^0
\end{bmatrix}
\begin{bmatrix}
\mathbf{x}_0^0 \\
\mathbf{x}_1^0 \\
\vdots \\
\mathbf{x}_n^0
\end{bmatrix}
=
\begin{bmatrix}
\mathbf{H}_0^0 \hat{\mathbf{u}}_0^0 \\
\mathbf{H}_1^0 \hat{\mathbf{u}}_1^0 \\
\vdots \\
\mathbf{H}_n^0 \hat{\mathbf{u}}_n^0
\end{bmatrix}
\tag{31}
$$

$$
\begin{bmatrix}
\mathbf{T}_{00}^0 & \mathbf{T}_{01}^0 & \cdots & \mathbf{T}_{0n}^0 \\
\mathbf{T}_{10}^0 & \mathbf{T}_{11}^0 & \cdots & \mathbf{T}_{1n}^0 \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{T}_{n0}^0 & \mathbf{T}_{n1}^0 & \cdots & \mathbf{T}_{nn}^0
\end{bmatrix}
\begin{bmatrix}
\mathbf{x}_0^0 \\
\mathbf{x}_1^0 \\
\vdots \\
\mathbf{x}_n^0
\end{bmatrix}
=
\begin{bmatrix}
\mathbf{H}_0^0 \hat{\mathbf{t}}_0^0 \\
\mathbf{H}_1^0 \hat{\mathbf{t}}_1^0 \\
\vdots \\
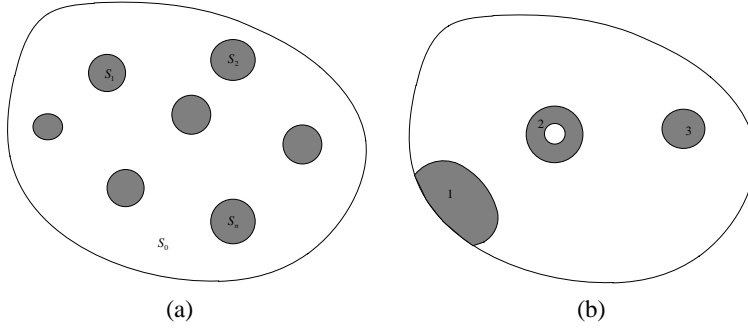\mathbf{H}_n^0 \hat{\mathbf{t}}_n^0
\end{bmatrix}
\tag{32}
$$

Figure 1: (a) The model of matrix with inclusions; (b) Three kinds of inclusions.

where scripts 0 and k, $k=1,\ldots, n$ indicate the subdomain of matrix and the k-th inclusion, respectively.

Three types of inclusions can be considered as shown in Figure 1(b) and the inclusion type 2 is totally in the matrix and it is hollow. Then for the general case, the system equations for the k-th inclusion domain can be written as

$$\begin{bmatrix} \mathbf{U}_{00}^k & \mathbf{U}_{0k}^k \\ \mathbf{U}_{k0}^k & \mathbf{U}_{kk}^k \end{bmatrix} \left\{ \begin{array}{c} \mathbf{x}_0^k \\ \mathbf{x}_k^k \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{H}_0^k \hat{\mathbf{u}}_0^k \\ \mathbf{H}_k^k \hat{\mathbf{u}}_k^k \end{array} \right\} \tag{33}$$

$$\begin{bmatrix} \mathbf{T}_{00}^k & \mathbf{T}_{0k}^k \\ \mathbf{T}_{k0}^k & \mathbf{T}_{kk}^k \end{bmatrix} \left\{ \begin{array}{c} \mathbf{x}_0^k \\ \mathbf{x}_k^k \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{H}_0^k \hat{\mathbf{t}}_0^k \\ \mathbf{H}_k^k \hat{\mathbf{t}}_k^k \end{array} \right\} \tag{34}$$

If the inclusions are solid and totally embed in the matrix (the inclusion type 3 in Figure 1(b)), then Equations (33) and (34) can be rewritten as

$$\mathbf{U}_{00}^k \mathbf{x}_0^k = \mathbf{H}_0^k \hat{\mathbf{u}}_0^k \tag{35}$$

$$\mathbf{T}_{00}^k \mathbf{x}_0^k = \mathbf{H}_0^k \hat{\mathbf{t}}_0^k \tag{36}$$

One can obtain the final system equation for the whole domain as

$$
\begin{bmatrix}
\mathbf{A}^0_{00} & \mathbf{A}^0_{01} & \cdots & \mathbf{A}^0_{0n} & 0 & 0 & 0 & 0 & 0 \\
\mathbf{U}^0_{10} & \mathbf{U}^0_{11} & \cdots & \mathbf{U}^0_{1n} & -\mathbf{U}^1_{00} & -\mathbf{U}^1_{01} & 0 & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & 0 & 0 & \ddots & 0 & 0 \\
\mathbf{U}^0_{n0} & \mathbf{U}^0_{n1} & \cdots & \mathbf{U}^0_{nn} & 0 & 0 & 0 & -\mathbf{U}^n_{00} & -\mathbf{U}^n_{0n} \\
\mathbf{T}^0_{10} & \mathbf{T}^0_{11} & \cdots & \mathbf{T}^0_{1n} & \mathbf{T}^1_{00} & \mathbf{T}^1_{01} & 0 & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & 0 & 0 & \ddots & 0 & 0 \\
\mathbf{T}^0_{n0} & \mathbf{T}^0_{n1} & \cdots & \mathbf{T}^0_{nn} & 0 & 0 & 0 & \mathbf{T}^n_{00} & \mathbf{T}^n_{0n} \\
0 & 0 & 0 & 0 & \mathbf{A}^1_{00} & \mathbf{A}^1_{01} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \ddots & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{A}^n_{00} & \mathbf{A}^n_{0n}
\end{bmatrix}
\begin{Bmatrix}
\mathbf{x}^0_0 \\ \mathbf{x}^0_1 \\ \vdots \\ \mathbf{x}^0_n \\ \mathbf{x}^1_0 \\ \mathbf{x}^1_1 \\ \vdots \\ \mathbf{x}^n_0 \\ \mathbf{x}^n_n
\end{Bmatrix}
=
\begin{Bmatrix}
\mathbf{d}^0_0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0 \\ \mathbf{d}^1_1 \\ \vdots \\ \mathbf{d}^n_n
\end{Bmatrix}
\tag{37}
$$

Equation (37) is the conventional multi-domain solver for the composites of matrix with inclusions. All the unknown parameters x about the matrix and inclusions are solved in this equation.

Now we derive the improved formulation for composites with inclusions, which is based on the multi-domain solver.

From Equation (31) one can have

$$
\mathbf{H}^0_k \hat{\mathbf{u}}^0_k = \mathbf{U}^0_{k0}\mathbf{x}^0_0 + \mathbf{U}^0_{k1}\mathbf{x}^0_1 + \cdots + \mathbf{U}^0_{kn}\mathbf{x}^0_n = \sum_{j=0}^{n} \mathbf{U}^0_{kj}\mathbf{x}^0_j
\tag{38}
$$

From Equation (33) one can have

$$
\left\{ \begin{matrix} \mathbf{x}^k_0 \\ \mathbf{x}^k_k \end{matrix} \right\} =
\begin{bmatrix} \mathbf{U}^k_{00} & \mathbf{U}^k_{0k} \\ \mathbf{U}^k_{k0} & \mathbf{U}^k_{kk} \end{bmatrix}^{-1}
\left\{ \begin{matrix} \mathbf{H}^k_0 \hat{\mathbf{u}}^k_0 \\ \mathbf{H}^k_k \hat{\mathbf{u}}^k_k \end{matrix} \right\} =
\begin{bmatrix} \mathbf{A}_U & \mathbf{B}_U \\ \mathbf{C}_U & \mathbf{D}_U \end{bmatrix}
\left\{ \begin{matrix} \mathbf{H}^k_0 \hat{\mathbf{u}}^k_0 \\ \mathbf{H}^k_k \hat{\mathbf{u}}^k_k \end{matrix} \right\}
\tag{39}
$$

where

$$
\begin{bmatrix} \mathbf{A}_U & \mathbf{B}_U \\ \mathbf{C}_U & \mathbf{D}_U \end{bmatrix} =
\begin{bmatrix} \mathbf{U}^k_{00} & \mathbf{U}^k_{0k} \\ \mathbf{U}^k_{k0} & \mathbf{U}^k_{kk} \end{bmatrix}^{-1}
\tag{40}
$$

Then from Equation (39) one can obtain

$$
\mathbf{x}^k_0 = \mathbf{A}_U \mathbf{H}^k_0 \hat{\mathbf{u}}^k_0 + \mathbf{B}_U \mathbf{H}^k_k \hat{\mathbf{u}}^k_k
\tag{41}
$$

And from Equation (33) one can obtain

$$
\mathbf{H}^k_k \hat{\mathbf{u}}^k_k = \mathbf{U}^k_{k0}\mathbf{x}^k_0 + \mathbf{U}^k_{kk}\mathbf{x}^k_k
\tag{42}
$$

At the interface between the matrix and inclusion, both the displacement and traction must be continuous. If we use the same set of nodes distributed on the interface, we can obtain

$$\mathbf{H}_0^k \hat{\mathbf{u}}_0^k = \mathbf{H}_k^0 \hat{\mathbf{u}}_k^0 \tag{43}$$

$$\mathbf{H}_0^k \hat{\mathbf{t}}_0^k = -\mathbf{H}_k^0 \hat{\mathbf{t}}_k^0 \tag{44}$$

Substituting Equations (42) and (43) into Equation (41) and using Equation (38), one can have

$$\mathbf{x}_0^k = \left[\mathbf{I} - \mathbf{B}_U \mathbf{U}_{k0}^k\right]^{-1} \left(\mathbf{A}_U \sum_{j=0}^{n} \mathbf{U}_{kj}^0 \mathbf{x}_j^0 + \mathbf{B}_U \mathbf{U}_{kk}^k \mathbf{x}_k^k\right) \tag{45}$$

where $\mathbf{I}$ is the identity operator.

Using the boundary conditions, for the matrix domain one can have

$$\begin{bmatrix} \mathbf{A}_{00}^0 & \mathbf{A}_{01}^0 & \cdots & \mathbf{A}_{0n}^0 \\ \mathbf{T}_{10}^0 & \mathbf{T}_{11}^0 & \cdots & \mathbf{T}_{1n}^0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{T}_{n0}^0 & \mathbf{T}_{n1}^0 & \cdots & \mathbf{T}_{nn}^0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_0^0 \\ \mathbf{x}_1^0 \\ \vdots \\ \mathbf{x}_n^0 \end{bmatrix} = \begin{bmatrix} \mathbf{d}_0^0 \\ \mathbf{H}_1^0 \hat{\mathbf{t}}_1^0 \\ \vdots \\ \mathbf{H}_n^0 \hat{\mathbf{t}}_n^0 \end{bmatrix} \tag{46}$$

For the k-th inclusion domain one can have

$$\begin{bmatrix} \mathbf{T}_{00}^k & \mathbf{T}_{0k}^k \\ \mathbf{A}_{k0}^k & \mathbf{A}_{kk}^k \end{bmatrix} \left\{ \begin{array}{c} \mathbf{x}_0^k \\ \mathbf{x}_k^k \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{H}_0^k \hat{\mathbf{t}}_0^k \\ \mathbf{d}_k^k \end{array} \right\} \tag{47}$$

Submitting Equation (45) into Equation (47) and using Equation (44) we can obtain

$$-\mathbf{H}_k^0 \hat{\mathbf{t}}_k^0 = \mathbf{T}_{00}^k \left[\mathbf{I} - \mathbf{B}_U \mathbf{U}_{k0}^k\right]^{-1} \mathbf{A}_U \sum_{j=0}^{n} \mathbf{U}_{kj}^0 \mathbf{x}_j^0$$
$$+ (\mathbf{T}_{00}^k \left[\mathbf{I} - \mathbf{B}_U \mathbf{U}_{k0}^k\right]^{-1} \mathbf{B}_U \mathbf{U}_{kk}^k + \mathbf{T}_{0k}^k)\mathbf{x}_k^k \tag{48}$$

$$\mathbf{d}_k^k = \mathbf{A}_{k0}^k \left[\mathbf{I} - \mathbf{B}_U \mathbf{U}_{k0}^k\right]^{-1} \mathbf{A}_U \sum_{j=0}^{n} \mathbf{U}_{kj}^0 \mathbf{x}_j^0$$
$$+ (\mathbf{A}_{k0}^k \left[\mathbf{I} - \mathbf{B}_U \mathbf{U}_{k0}^k\right]^{-1} \mathbf{B}_U \mathbf{U}_{kk}^k + \mathbf{A}_{kk}^k)\mathbf{x}_k^k \tag{49}$$

Then by assembling Equations (48) and (49) into Equation (46) one can obtain the final system equation as

$$\begin{bmatrix} \mathbf{A}_{00}^0 & \mathbf{A}_{01}^0 & \cdots & \mathbf{A}_{0n}^0 & 0 & 0 & 0 \\ \mathbf{F}_{10}^0 & \mathbf{F}_{11}^0 & \cdots & \mathbf{F}_{1n}^0 & \mathbf{G}^1 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & 0 & \ddots & 0 \\ \mathbf{F}_{n0}^0 & \mathbf{F}_{n1}^0 & \cdots & \mathbf{F}_{nn}^0 & 0 & 0 & \mathbf{G}^n \\ \mathbf{Q}_{10}^0 & \mathbf{Q}_{11}^0 & \cdots & \mathbf{Q}_{1n}^0 & \mathbf{R}^1 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & 0 & \ddots & 0 \\ \mathbf{Q}_{n0}^0 & \mathbf{Q}_{n1}^0 & \cdots & \mathbf{Q}_{nn}^0 & 0 & 0 & \mathbf{R}^n \end{bmatrix} \left\{ \begin{array}{c} \mathbf{x}_0^0 \\ \mathbf{x}_1^0 \\ \vdots \\ \mathbf{x}_n^0 \\ \mathbf{x}_1^1 \\ \vdots \\ \mathbf{x}_n^n \end{array} \right\} = \left\{ \begin{array}{c} \mathbf{d}_0^0 \\ 0 \\ \vdots \\ 0 \\ \mathbf{d}_1^1 \\ \vdots \\ \mathbf{d}_n^n \end{array} \right\} \tag{50}$$

where

$$\mathbf{F}_{kj}^0 = \mathbf{T}_{kj}^0 + \mathbf{E}_1^k \mathbf{U}_{kj}^0 \tag{51}$$

$$\mathbf{G}^k = \mathbf{E}_2^k \mathbf{U}_{kk}^k + \mathbf{T}_{0k}^k \tag{52}$$

$$\mathbf{Q}_{kj}^0 = \mathbf{E}_3^k \mathbf{U}_{kj}^0 \tag{53}$$

$$\mathbf{R}^k = \mathbf{E}_4^k \mathbf{U}_{kk}^k + \mathbf{A}_{kk}^k \tag{54}$$

$$\mathbf{E}_1^k = \mathbf{T}_{00}^k \left[\mathbf{I} - \mathbf{B}_U \mathbf{U}_{k0}^k \right]^{-1} \mathbf{A}_U \tag{55}$$

$$\mathbf{E}_2^k = \mathbf{T}_{00}^k \left[\mathbf{I} - \mathbf{B}_U \mathbf{U}_{k0}^k \right]^{-1} \mathbf{B}_U \tag{56}$$

$$\mathbf{E}_3^k = \mathbf{A}_{k0}^k \left[\mathbf{I} - \mathbf{B}_U \mathbf{U}_{k0}^k \right]^{-1} \mathbf{A}_U \tag{57}$$

$$\mathbf{E}_4^k = \mathbf{A}_{k0}^k \left[\mathbf{I} - \mathbf{B}_U \mathbf{U}_{k0}^k \right]^{-1} \mathbf{B}_U \tag{58}$$

For the case when the inclusions are solid and totally embed in the matrix, Equation (50) can be rewritten as

$$\begin{bmatrix} \mathbf{A}_{00}^0 & \mathbf{A}_{01}^0 & \cdots & \mathbf{A}_{0n}^0 \\ \mathbf{D}_{10}^0 & \mathbf{D}_{11}^0 & \cdots & \mathbf{D}_{1n}^0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{D}_{n0}^0 & \mathbf{D}_{n1}^0 & \cdots & \mathbf{D}_{nn}^0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_0^0 \\ \mathbf{x}_1^0 \\ \vdots \\ \mathbf{x}_n^0 \end{bmatrix} = \begin{bmatrix} \mathbf{d}_0^0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{59}$$

where

$$\mathbf{D}_{kj}^0 = \mathbf{T}_{kj}^0 + \mathbf{C}^k \mathbf{U}_{kj}^0 \tag{60}$$

$$\mathbf{C}^k = \mathbf{T}_{00}^k \left[\mathbf{U}_{00}^k \right]^{-1} \tag{61}$$

If the inclusions are of the same size and material, the corresponding incidence matrices $\mathbf{C}^k$ are also identical. Therefore, $\mathbf{C}^k$ is required to be formed only once. If the inclusions are of the same shape, $\mathbf{C}^k$ can also be calculated only once because of the similar relationship between inclusion phases of different size and material. If we assume the Poisson's ratios of the inclusions are the same, one can have the following relationship

$$\mathbf{C}^{k+1} = \frac{E^{k+1}}{E^k} \frac{r^k}{r^{k+1}} \mathbf{C}^k \tag{62}$$

where E^k and r^k are Young's module and radius of the k-th inclusion, respectively.

In Equations (50) and (59), the unknowns on the interfaces are assembled only once compared with Equation (37), thus, the total unknowns in the final system equation can be reduced by the improved multi-domain solver. In equation (50), if the inclusions are of the same shape and the same boundary condition, Equations (52), (54), (55), (56), (57) and (58) can also be computed once and the computational resource, both the time and memory required, can be saved further. Compared with the simplified approach proposed by Zhang, Tanaka and Matsumoto (2004b), no assumption is needed and it can be applied to analyze more general composites. The improved multi-domain solver is also very suitable to couple with FMM for large scale computation.

5   Numerical examples

The proposed technique has been implemented in C++ program. Three numerical examples are studied in this section. For the purpose of error estimation, a formula is defined as

$$e = \frac{1}{|u|_{\max}} \sqrt{\frac{1}{N} \sum_{i=1}^{N} (u_i^{(e)} - u_i^{(n)})^2} \tag{63}$$

where $u_i^{(e)}$ and $u_i^{(n)}$ refer to the exact and numerical solutions respectively and $|u|_{\max}$ is the maximum value of u over N nodes.

In all the examples, Gaussian elimination method is used to solve the final system equations and both the multi-domain solver and improved multi-domain solver are used.

5.1   Validation of the method

In this example, a hollow sphere shown in Figure 2 is considered, the outer radius of the sphere is 10 and the inner radius is 1, the Young's module $E = 1.0$ and the Poisson's ratio $v = 0.25$ Direchlet boundary conditions on all the surfaces are imposed, according to the following exact solution:

$$u_x = \frac{2x + y + z}{2} \tag{64}$$

$$u_y = \frac{x + 2y + z}{2} \tag{65}$$

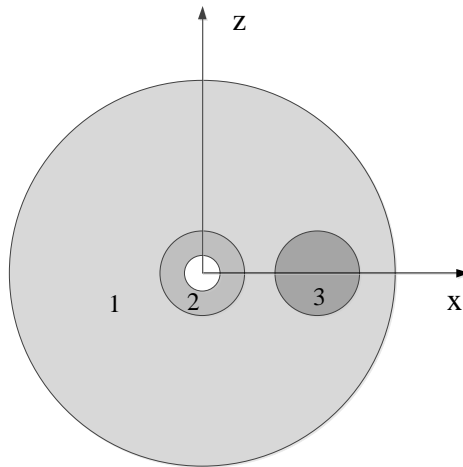$$u_z = \frac{x + y + 2z}{2} \tag{66}$$

Figure 2: A hollow sphere with three subdomains.

In order to apply the multi-domain and improved multi-domain solver, the hollow sphere is divided into three subdomains with the same materials, see Figure 2. For the subdomain 2, the outer radius is 2 and the inner radius is 1. For subdomain 3, the radius is 2 and its center located at point (6, 0, 0).

In the model, the spherical face is considered as one face and all the spherical faces have the same number of nodes distributed on them. Several different node arrangements are used to show the efficiency and accuracy of the technique. Figure 3 shows the relative error of u_x computed by the nodes on line (1, 0, 0) to (10, 0, 0). From Figure 3 one can observe that the improved multi-domain solver has the same precision as the conventional multi-domain solver. The CPU time for solving the final system equation is shown in Figure 4 and we can observe that the improved multi-domain is more efficient than the conventional multi-domain solver since it can reduce the total DOFs in the final system equation.

5.2   *Cube with eight spherical inclusions*

A cube with eight spherical inclusions shown in Figure 5 is considered in this example. The parameters of the cube are: the length *a*=2, the Young's module $E = 1.0$, Poisson's ratio $v = 0.25$. The cube is roller supported on the left end and a uniform distributed traction is loaded on outer normal direction of the right end. These boundary conditions allow us to estimate the effective Young's module in the axial
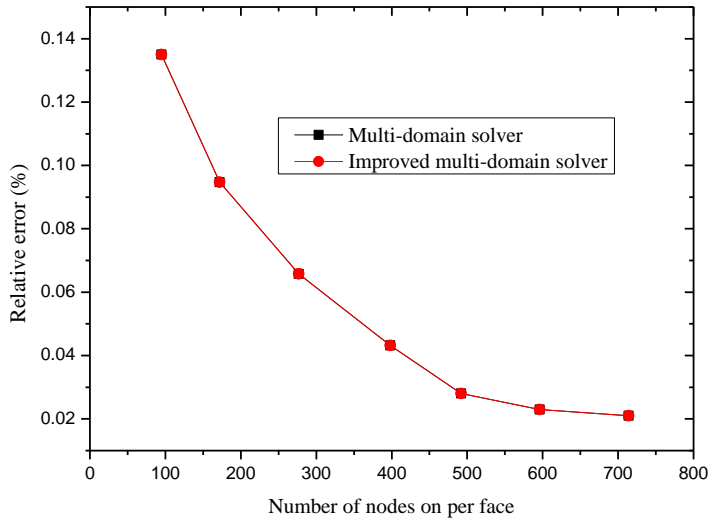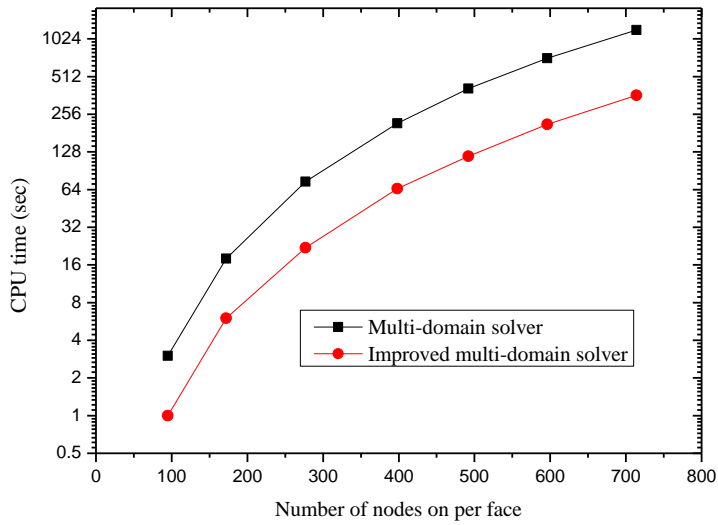
Figure 3: Relative error.

Figure 4: CPU time.

direction as

$$E_{eff} = \frac{\sigma L}{\Delta u} \qquad (67)$$

where E_{eff} is the estimated effective Young's module, L is the length, σ is average stress of the two ends and Δu is the difference of displacement between the two ends.

Two models are considered. In the first model, the parameters of the inclusions are: the radius r=0.2 and the Young's module E_{in} changes from 1 to 10. In the second model, the Young's module $E_{in} = 8$ and the radius of the inclusion changes from 0.1 to 0.4. In all the models, the Poisson's ratio $v = 0.25$, 600 nodes are distributed on the outer faces and 172 nodes are distributed on each interface. Table 1 shows the information of total unknowns in the final system equation and the computational time. From Table 1, one can indicate that the number of DOFs in the improved multi-domain solver is reduced to nearly one half of that in the conventional multi-domain solver. Thus, the improved multi-domain solver is more efficient and From Table 1, one can observe that the CPU time has been saved much by improve solver. Figure 6 and Figure 7 show the effective Young's modules while the Young's module of the inclusion and radius of the inclusion changes, respectively. In order to show the validation of the method, the Mori-Tanaka method [Karris (1989)] is also used. From Figure 6 and Figure 7 one can observe that the results obtained by the improved multi-domain solver and conventional multi-domain solver are the same and both of them have good agreement with the Mori-Tanaka method. It should be noted that Mori-Tanaka method can only be used to estimate the effective module of composite for some special cases, it cannot give the field of stress or displacement under boundary conditions. However, by applying numerical methods like introduced in this paper, these aims can be easily achieved.

If the matrix has more inclusions, the Hybrid BNM can also be applied easily. However, without fast algorithms the technique cannot solve large scale problems [Wang, Miao and Zheng (2010); Wang, Miao, Zhu and Zhang (2012); Wang, Miao and Zhu (2013a); Wang, Miao and Zhu (2013c); Miao, Wang, Zhu and Li (2014)]

Table 1: Time information of cub with 8 spherical inclusions.

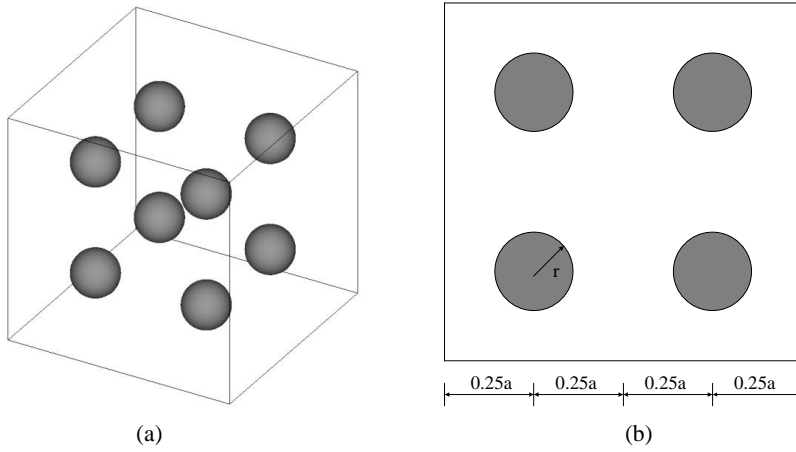| Solver type	Total unknowns in the final equation	CPU time (s)
Multi-domain solver	10056	610
Improved multi-domain solver	5928	124

(a)   (b)

Figure 5: Model of cub with 8 uniformly distributed spherical inclusions.
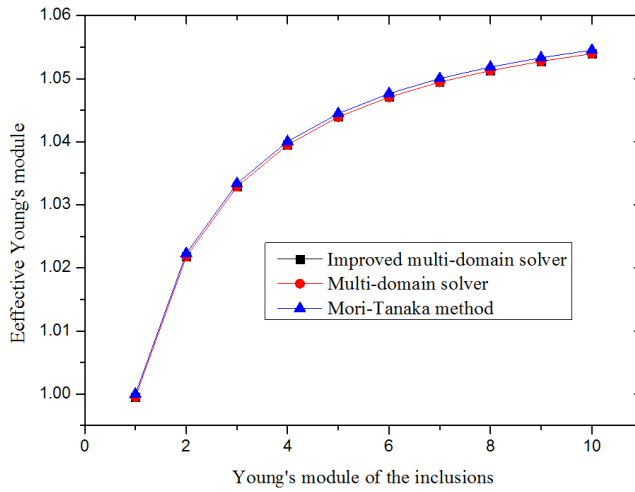

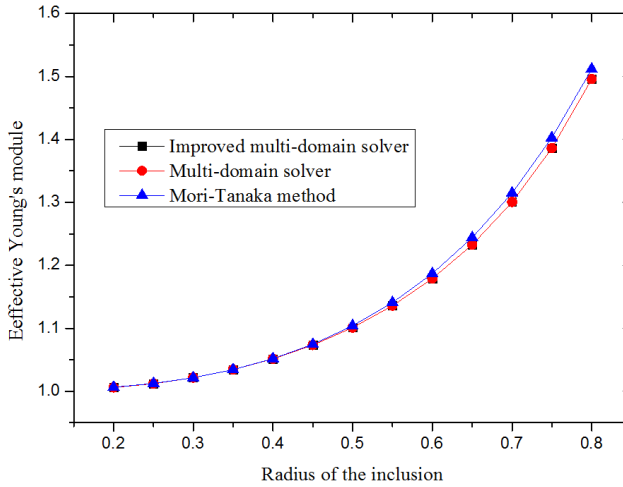Figure 6: Effective Young's module with Young's module of the inclusion.

Figure 7: Effective Young's module with radius of the inclusion.

5.3   *A square RVE*

A square RVE model with an inclusion embedded is considered in this example. The geometry is shown in Figure 8. The parameters of the matrix are: the length L=100, the width $W = 20$, the height $H = 20$, the Young's module $E_{matrix} = 1.0$ and the Poisson's ratio $v = 0.25$. The RVE is roller supported on one end and applied with a uniform normal displacement $u_y = 100$ on the other end. The parameters of the inclusion are: the length L_c=50, the outer radius r=5, the thickness t=0.4 and the Poisson's ratio $v = 0.25$. If the Young's module of the inclusion is supposed to be much higher than the matrix, this model can be considered as a CNT based composites.

The ratio of the Young's modules between the inclusion and the matrix is defined as $c = E_{inclusion}/E_{matrix}$. Numerical results for the displacements at lines $x = 0, z = 4.8$ (through the inclusion) and $x = 0, z = 5.1$ (very near the inclusion) with different values of ratio c are presented in Figure 9. In this example, we use *MS* and *IMS* denote the *multi-domain solver* and *improved multi-domain solver*, respectively. The results obtained by the multi-domain solver and improved multi-domain solver are the same. From Figure 8, we can observe that the displacements at the two lines are very close with a given ratio c. The difference of the displacements within the inclusion (r=4.8) and near the inclusion (r=5.1) becomes smaller as the increase of
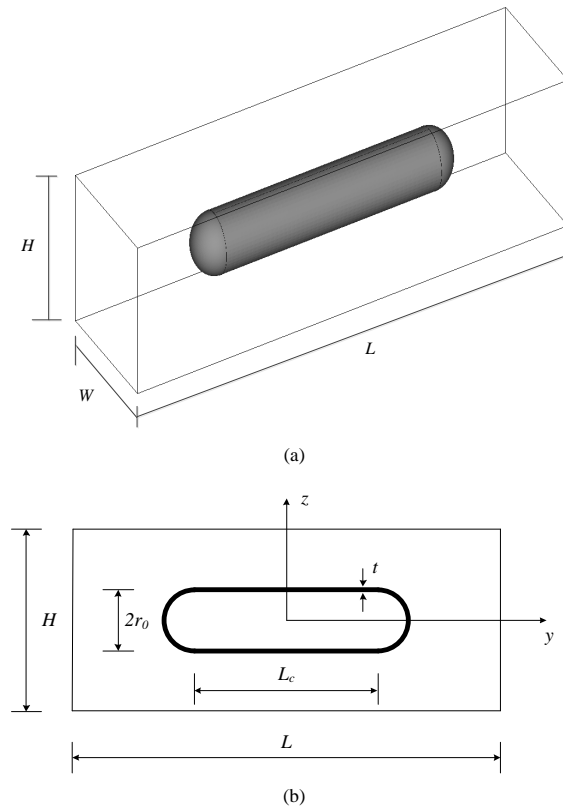
(a)

(b)

Figure 8: A square RVE model.

ratio c. The displacements become almost constant along the length of the inclusion when c=1000 and the results are very similar to a RVE with rigid inclusion.

Figure 10 shows the displacements at line $x = 0, z = 4.8$ with four different thickness t of the inclusion while c=200. While $t = 5.0$, the inclusion is solid and only its outer face is needed. In Figure 10, we can observe that thickness of the inclusion has little influence on the displacements at the evaluated line. One of the reasons is that the inclusion is nearly a rigid body in this case and the displacements of the inclusion is the same. Figure 11 shows the effective Young's module of the RVE with various thickness t and ratio of the Young's modules between the inclusion and the matrix. From Figure 11 one can observe that the difference of the effective Young's modules becomes littler and littler as the increase of the ratio c and the effective Young's modules become very close for the various thickness t when the ratio c is more than 6000. From this example we can find out that the thickness of

the inclusion has little influence on the effective Young's module of the RVE when the Young's module of the inclusion is much higher than the matrix, and only the outer face of the inclusion is needed be modeled, which can reduce much resource further.
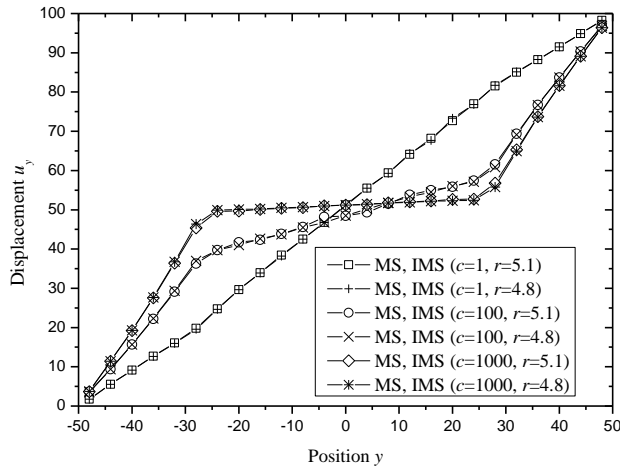

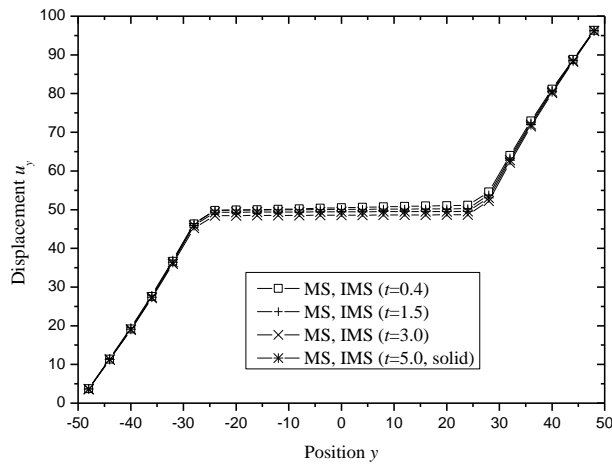Figure 9: Displacements along the axial lines (t=0.4).

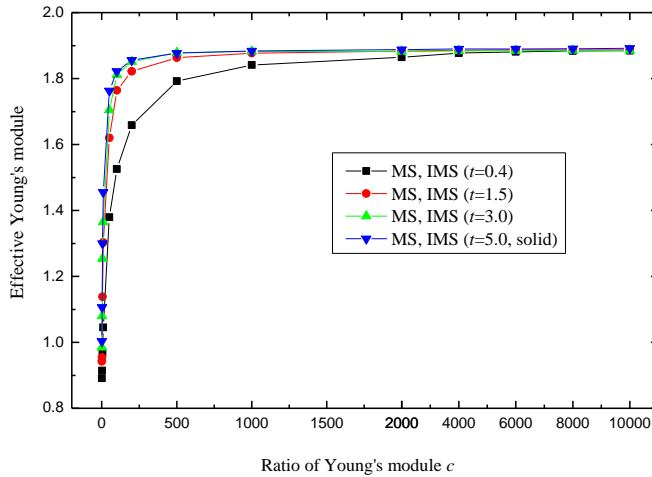Figure 10: Displacements along the axial lines (c=2000).

Figure 11: Effective Young's module of the RVE with various *t* and *c*.

6 Conclusions

In this paper, an improved multi-domain solver is proposed for the mechanical analysis of composites with inclusions by the Hybrid BNM. Continuity conditions are used in the improved multi-domain solver as the conventional multi-domain solver. The unknown vectors on the interfaces are needed to be assembled only once in the final system equation of improved solver, thus, the computational time and memory required can be reduced.

The improved multi-domain model proposed can be applied to more general composites than the simplified approach, and especial suitable for the composites when the inclusions are solid and totally embedded in the matrix domain. Composites which similar as rigid inclusion based composites are also discussed in the paper and we can find out that the thickness of the rigid inclusion has little influence on the effective Young's module.

The improved multi-domain solver is very suitable for large-scale computation and coupling it with fast algorithm, such as FMM is a subject of future research.

The difference between the numerical and experimental results has been observe in the work of Yao, Xu, Wang and Zheng (2009) It means that the numerical method has to be improved by considering more practical microstructural factors, including the imperfect interface conditions and thermal contact between fibers. These researches will be investigated in the future.

References

Atluri, S.; Zhu, T. (1998): A new meshless local Petrov-Galerkin (MLPG) approach in computational mechanics. *Computational mechanics*, vol. 22, pp. 117-127.

Belytschko, T.; Lu, Y. Y.; Gu, L. (1994): Element-free Galerkin methods. *International journal for numerical methods in engineering*, vol. 37, pp. 229-256.

Bishay, P.; Atluri, S. (2012): High-Performance 3D Hybrid/Mixed, and Simple 3D Voronoi Cell Finite Elements, for Macro-& Micro-mechanical Modeling of Solids, Without Using Multi-field Variational Principles. *Computer Modeling in Engineering & Sciences(CMES)*, vol. 84, pp. 41-97.

Bishay, P. L.; Sladek, J.; Sladek, V.; Atluri, S. N. (2012): Analysis of Functionally Graded Magneto-Electro-Elastic Composites Using Hybrid/Mixed Finite Elements and Node-Wise Material Properties. *CMC: Computers, Materials & Continua*, vol. 29, pp. 213-262.

Chen, X.; Liu, Y. (2005): An advanced 3D boundary element method for characterizations of composite materials. *Engineering analysis with boundary elements*, vol. 29, pp. 513-523.

De, S.; Hong, J.-W.; Bathe, K. (2003): On the method of finite spheres in applications: towards the use with ADINA and in a surgical simulator. *Computational Mechanics*, vol. 31, pp. 27-37.

Dong, L.; Alotaibi, A.; Mohiuddine, S.; Atluri, S. (2014): Computational methods in engineering: a variety of primal & mixed methods, with global & local interpolations, for well-posed or ill-Posed BCs. *CMES: Computer Modeling in Engineering & Sciences*, vol. 99, pp. 1-85.

Dong, L.; Atluri, S. N. (2013a): Fracture & fatigue analyses: SGBEM-FEM or XFEM? Part 1: 2D structures. *CMES: Computer Modeling in Engineering & Sciences*, vol. 90, pp. 91-146.

Dong, L.; Atluri, S. N. (2013b): Fracture & fatigue analyses: SGBEM-FEM or XFEM? Part 2: 3D solids. *CMES: Computer Modeling in Engineering & Sciences*, vol. 90, pp. 379-413.

Dong, L.; Atluri, S. N. (2013c): SGBEM Voronoi Cells (SVCs), with embedded arbitrary-shaped inclusions, voids, and/or cracks, for micromechanical modeling

of heterogeneous materials. *CMC: Computers, Materials & Continua*, vol. 33, pp. 111-154.

Dong, L.; El-Gizawy, A. S.; Juhany, K. A.; Atluri, S. N. (2014): A simple locking-alleviated 4-node mixed-collocation finite element with over-integration, for homogeneous or functionally-graded or thick-section laminated composite beams. *CMC: Computers, Materials & Continua*, vol. 40, pp. 49-77.

Eischen, J.; Torquato, S. (1993): Determining elastic behavior of composites by the boundary element method. *Journal of applied physics*, vol. 74, pp. 159-170.

Gingold, R. A.; Monaghan, J. J. (1977): Smoothed particle hydrodynamics-theory and application to non-spherical stars. *Monthly notices of the royal astronomical society*, vol. 181, pp. 375-389.

Gu, Y.; Liu, G. (2002): A boundary point interpolation method for stress analysis of solids. *Computational Mechanics*, vol. 28, pp. 47-54.

Hou, T. Y.; Wu, X.-H. (1997): A multiscale finite element method for elliptic problems in composite materials and porous media. *Journal of computational physics*, vol. 134, pp. 169-189.

Kari, S.; Berger, H.; Rodriguez-Ramos, R.; Gabbert, U. (2007): Computational evaluation of effective material properties of composites reinforced by randomly distributed spherical particles. *Composite structures*, vol. 77, pp. 223-231.

Karris, A. (1989): An Examination of the Mori-Tanaka Effective Medium–Approximation for Multiphase Composites. *Journal of Applied Mechanics*, vol. 56, pp. 83-88.

Li, X.; Zhu, J. (2009): A Galerkin boundary node method and its convergence analysis. *Journal of computational and applied mathematics*, vol. 230, pp. 314-328.

Liszka, T.; Duarte, C.; Tworzydlo, W. (1996): hp-Meshless cloud method. *Computer Methods in Applied Mechanics and Engineering*, vol. 139, pp. 263-288.

Liu, W. K.; Jun, S.; Zhang, Y. F. (1995): Reproducing kernel particle methods. *International Journal for Numerical Methods in Fluids*, vol. 20, pp. 1081-1106.

Liu, Y.; Nishimura, N.; Otani, Y. (2005): Large-scale modeling of carbon-nanotube composites by a fast multipole boundary element method. *Computational Materials Science*, vol. 34, pp. 173-187.

Liu, Y.; Nishimura, N.; Otani, Y.; Takahashi, T.; Chen, X.; Munakata, H. (2005): A fast boundary element method for the analysis of fiber-reinforced composites based on a rigid-inclusion model. *Journal of applied mechanics*, vol. 72, pp. 115-128.

Lucy, L. B. (1977): A numerical approach to the testing of the fission hypothesis.

The astronomical journal, vol. 82, pp. 1013-1024.

Miao, Y.; Wang, Q.; Liao, B.; Zheng, J. (2009): A dual hybrid boundary node method for 2D elastodynamics problems. *Computer Modeling in Engineering and Sciences (CMES)*, vol. 53, pp. 1.

Miao, Y.; Wang, Q.; Zhu, H.; Li, Y. (2014): Thermal analysis of 3D composites by a new fast multipole hybrid boundary node method. *Computational Mechanics*, vol. 53, pp. 77-90.

Miao, Y.; Wang, Y.-h. (2006): Meshless analysis for three-dimensional elasticity with singular hybrid boundary node method. *Applied Mathematics and Mechanics*, vol. 27, pp. 673-681.

Miao, Y.; Wang, Y.; Wang, Y. (2009): A meshless hybrid boundary-node method for Helmholtz problems. *Engineering analysis with boundary elements*, vol. 33, pp. 120-127.

Miao, Y.; Wang, Y.; Yu, F. (2005): Development of hybrid boundary node method in two-dimensional elasticity. *Engineering analysis with boundary elements*, vol. 29, pp. 703-712.

Mukherjee, Y. X.; Mukherjee, S. (1997): The boundary node method for potential problems. *International Journal for Numerical Methods in Engineering*, vol. 40, pp. 797-815.

Nishimura, N.; Liu, Y. (2004): Thermal analysis of carbon-nanotube composites using a rigid-line inclusion model by the boundary integral equation method. *Computational Mechanics*, vol. 35, pp. 1-10.

Rokhlin, V. (1985): Rapid solution of integral equations of classical potential theory. *Journal of Computational Physics*, vol. 60, pp. 187-207.

Segurado, J.; LLorca, J. (2004): A new three-dimensional interface finite element to simulate fracture in composites. *International journal of solids and structures*, vol. 41, pp. 2977-2993.

Singh, I.; Tanaka, M.; Endo, M. (2007): Thermal analysis of CNT-based nano-composites by element free Galerkin method. *Computational Mechanics*, vol. 39, pp. 719-728.

Wang, H.; Yao, Z. (2005): A new fast multipole boundary element method for large scale analysis of mechanical properties in 3D particle-reinforced composites. *Computer Modeling in Engineering & Sciences*, vol. 7, pp. 85-95.

Wang, H.; Yao, Z. (2008): A rigid-fiber-based boundary element model for strength simulation of carbon nanotube reinforced composites. *CMES: Computer Modeling in Engineering & Sciences*, vol. 29, pp. 1-13.

Wang, Q.; Miao, Y.; Zheng, J. (2010): The hybrid boundary node method accel-

erated by fast multipole expansion technique for 3D elasticity. *Computer Modeling in Engineering and Sciences*, vol. 70, pp. 123-151.

Wang, Q.; Miao, Y.; Zhu, H. (2013a): A fast multipole hybrid boundary node method for composite materials. *Computational Mechanics*, vol., pp. 885-897.

Wang, Q.; Miao, Y.; Zhu, H. (2013b): A new formulation for thermal analysis of composites by hybrid boundary node method. *International Journal of Heat and Mass Transfer*, vol. 64, pp. 322-330.

Wang, Q.; Miao, Y.; Zhu, H. (2013c): Numerical Simulation of Heat Conduction Problems by a New Fast Multipole Hybrid Boundary-Node Method. *Numerical Heat Transfer, Part B: Fundamentals*, vol. 64, pp. 436-459.

Wang, Q.; Miao, Y.; Zhu, H.; Zhang, C. (2012): An O (N) Fast Multipole Hybrid Boundary Node Method for 3D Elasticity. *Computers Materials and Continua*, vol. 28, pp. 1-25.

Wang, Q.; Zheng, J.; Miao, Y.; Lv, J. (2011): The multi-domain hybrid boundary node method for 3D elasticity. *Engineering Analysis with Boundary Elements*, vol. 35, pp. 803-810.

Yao, Z.-H.; Xu, J.-D.; Wang, H.-T.; Zheng, X.-P. (2009): Simulation of CNT composites using fast multipole BEM. *Journal of Marine Science and Technology*, vol. 17, pp. 194-202.

Zhang, J.; Qin, X.; Han, X.; Li, G. (2009): A boundary face method for potential problems in three dimensions. *International journal for numerical methods in engineering*, vol. 80, pp. 320-337.

Zhang, J.; Tanaka, M. (2008): Fast HdBNM for large-scale thermal analysis of CNT-reinforced composites. *Computational Mechanics*, vol. 41, pp. 777-787.

Zhang, J.; Tanaka, M.; Matsumoto, T. (2004a): Meshless analysis of potential problems in three dimensions with the hybrid boundary node method. *International Journal for Numerical Methods in Engineering*, vol. 59, pp. 1147-1166.

Zhang, J.; Tanaka, M.; Matsumoto, T. (2004b): A simplified approach for heat conduction analysis of CNT-based nano-composites. *Computer methods in applied mechanics and engineering*, vol. 193, pp. 5597-5609.

Zhang, J.; Yao, Z. (2001): Meshless regular hybrid boundary node method. *CMES-Computer Modeling in Engineering and Sciences*, vol. 2, pp. 307-318.

Zhang, J.; Yao, Z. (2004): The regular hybrid boundary node method for three-dimensional linear elasticity. *Engineering analysis with boundary elements*, vol. 28, pp. 525-534.

Zhang, J.; Yao, Z.; Li, H. (2002): A hybrid boundary node method. *International Journal for Numerical Methods in Engineering*, vol. 53, pp. 751-763.