Comput Mech
DOI 10.1007/s00466-013-0893-0

ORIGINAL PAPER

Thermal analysis of 3D composites by a new fast multipole hybrid boundary node method

Yu Miao · Qiao Wang · Hongping Zhu · Yinping Li

Received: 15 March 2013 / Accepted: 23 June 2013

Abstract This paper applies the hybrid boundary node method (Hybrid BNM) for the thermal analysis of 3D composites. A new formulation is derived for the inclusion-based composites. In the new formulation, the unknowns of the interfaces are assembled only once in the final system equation, which can reduce nearly one half of degrees of freedom (DOFs) compared with the conventional multi-domain solver when there are lots of inclusions. A new version of the fast multipole method (FMM) is also coupled with the new formulation and the technique is applied to thermal analysis of composites with many inclusions. In the new fast multipole hybrid boundary node method (FM-HBNM), a diagonal form for translation operators is used and the method presented can be applied to the computation of more than 1,000,000 DOFs on a personal computer. Numerical examples are presented to analyze the thermal behavior of composites with many inclusions.

Keywords Hybrid boundary node method · 3D composites · New formulation · Fast multipole method · New FM-HBNM

1 Introduction

Composites are being increasingly applied to many industrial projects, which leads to much interest and research in the numerical simulation of these materials. In order to analyze the thermal properties of composite materials, many numerical models based on the finite element method (FEM) [1,2] and boundary element method (BEM) [3,4] have been developed. BEM is one of the mostly investigated methods for the analysis of thermal problems of composites in recent years.

Although the FEM and BEM are widely investigated and have been applied into many areas, they have much difficulty in solving problems involving changing domains such as large deformation or crack propagation. Besides, the task of mesh generation for complex geometries is often time-consuming and prone to errors by FEM or BEM. This may become worse when deal with composite materials. In order to overcome these disadvantages of methods based on cells, a new class of numerical methods, namely, the meshless or meshfree methods have been proposed and developed in recent years. Many kinds of meshless methods have been proposed so far, including the element free Galerkin method (EFG) [5], the meshless local Petrov–Galerkin (MLPG) approach [6], the boundary node method (BNM) [7], the Galerkin boundary node method (GBNM) [8], the boundary face method (BFM) [9] and the hybrid boundary node method (Hybrid BNM), etc.

The Hybrid BNM was proposed by Zhang et al. [10,11], which combines the MLS approximation scheme with the hybrid displacement variational formula. This method has been developed by Miao et al. [12,13] and applied to elastodynamics problems [14], Helmholtz problems [15] and multi-domain problems [16]. The Hybrid BNM not only reduce the spatial dimensions by one like BEM or BNM, but also does not require boundary element meshes, neither for the purpose of interpolation of the solution variables nor for the integration of energy. In fact, the Hybrid BNM requires only discrete node located on the surface of the domain and its parametric representation.

Y. Miao · Q. Wang (✉) · H. Zhu
School of Civil Engineering and Mechanics, Huazhong University of Science and Technology, Wuhan 430074, China
e-mail: hust.qiaowang@gmail.com

Y. Li
State Key Laboratory of Geomechanics and Geotechnical Engineering, Institute of Rock and Soil Mechanics, The Chinese Academy of Sciences, Wuhan 430071, China

Published online: 13 July 2013

The meshless methods are very suitable for the simulation of composite materials since they can avoid the mesh task. However, few meshless methods have been applied to the analysis of composites so far. A simplified model based on the Hybrid BNM for the heat conduction analysis of carbon nanotubes (CNTs) based nano-composites was proposed by Zhang et al. [17]. In the simplified model, the host polymer is the only domain which is modeled, while the CNTs are treated as heat superconductors with constant and unknown temperatures constrained at their surfaces. Singh et al. [18] applied the EFG to the thermal analysis of CNT based composites with both the mulit-domain solver and simplified approach. In the simplified approach of Zhang and Singh, since the inclusions are treated as heat superconductors, it can not be applied to general composites, even though it can reduce the total degrees of freedom (DOFs) substantially. Multi-domain solver [16] can be used to overcome this restriction as the matrix and inclusions are modeled as separate regions. However, it leads much more computational time than the simplified model, since the total DOFs containing both the unknowns in matrix and inclusions.

In this paper, a new formulation based on the Hybrid BNM is derived for the thermal analysis of composites with inclusions. The new formulation can be applied to general composites like the multi-domain solver since no assumption is used. It can also reduce the total DOFs as the simplified approach and only the domain of the matrix needs to be modeled. In the new formulation, the final system equation has a dense and unsymmetrical matrix, which requires $O(N^2)$ memory and $O(N^3)$ operations if using the direct solvers, such as Gaussian elimination method, where N is the total unknowns. The computational time of an iterative solver is still $O(M \times N^2)$, where M is the number of iterations, and the memory required is also $O(N^2)$. Thus, it is impossible to apply the new formulation to simulate composites with lots of inclusions because of the large scale computation. The fast multipole method (FMM) [19,20] is one of the most widely investigated and applied methods for accelerating. It was introduced by Rokhlin [19] as a fast solution method for integral equations for two dimensional Laplace's equation. The computational cost for an iterative solver can be reduced from $O(N^2)$ to $O(N)$ by the FMM, which makes large scale computation possible.

Appling the FMM to accelerate the hybrid BNM for 3D potential problems was presented by Zhang et al. [21], called as FM-HBNM. Zhang also used the FM-HBNM to study the thermal behavior of CNT composites [22,23]. Wang et al. [24,25] developed the FM-HNBM for 3D elasticity problems and applied it to simulate the mechanical properties of composites [26] by coupling with the multi-domain solver. The FMM also have some implementations in some other meshless methods [27,28].

In the FMM, the major obstacle in achieving reasonable efficiency with high accuracy is the large number of the multipole to local (M2L) translations. In 3D case, the computational cost for M2L is proportional to $O(189p^4)$ in the worst case, where p is the number of terms in the truncated expansion series. To overcome this obstacle, Greengard and Rokhlin [29] proposed a new diagonal form, which can reduce the M2L cost to $O(p^3)$. Applying this new version of FMM to accelerate BEM in 3D potential problems has been investigated by some researchers [30–33].

This paper presents a new FM-HBNM for the thermal analysis of composites with inclusions. In the new FM-HBNM, the new diagonal form is used for translation operators and the computational cost of M2L translation can be further reduced to $O(p^3)$. A new formulation is also proposed for the thermal analysis of composites with inclusions and accelerated by the new FMM. It can be applied to general composites like the conventional multi-domain solver and the total DOFs can also be reduced as the simplified approach since the unknowns on the interfaces are computed only once in the final system equation.

This paper is organized as follows. In the Sect. 2, the Hybrid BNM for 3D heat conduction problems is reviewed. Then a new formulation is derived for the thermal analysis of composites with inclusions in the Sect. 3. The Sect. 4 gives the details of the new FM-HNBM. Finally, numerical examples are given.

2 The hybrid boundary node method

In this section, the conventional Hybrid BNM [11] is summarized. A steady state heat conduction problem is governed by Laplace's equation with proper boundary conditions. The Hybrid BNM is based on a modified variational principle. In 3D heat conduction problems, the functions in the modified variational principle assumed to be independent are: temperature T inside the domain, boundary temperature $\tilde{T}$ and boundary normal heat flux $\tilde{q}$. Consider a domain Ω enclosed by $\Gamma = \Gamma_T + \Gamma_q$ with $\bar{T}$ and $\bar{q}$ are the prescribed temperature and normal heat flux, respectively.

The $\tilde{T}$ and $\tilde{q}$ at the boundary Γ are approximated by the MLS approximation:

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

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

where N is the number of boundary nodes; $\hat{T}_J$ and $\hat{q}_J$ are nodal values, and $\Phi_J(\mathbf{s})$ is the shape function of the MLS approximation, corresponding to node $\mathbf{s}_J$.

The T and q inside the domain can be approximated by fundamental solutions as

$$T = \sum_{J=1}^{N} T_J^s x_J \tag{3}$$

$$q = \sum_{J=1}^{N} q_J^s x_J \tag{4}$$

where T_J^s is the fundamental solution; x_J is unknown parameter and

$$T_J^s = \frac{1}{\kappa} \frac{1}{4\pi r(Q, \mathbf{s}_J)} \tag{5}$$

$$q_J^s = -\kappa \frac{\partial T_J^s}{\partial n} \tag{6}$$

where κ is the heat conductivity, Q is a field point, $r(Q, \mathbf{s}_J)$ is the distance between the point Q and the node $\mathbf{s}_J$.

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. The following set of equations, expressed in matrix form, are given as

$$\mathbf{U}\mathbf{x} = \mathbf{H}\hat{\mathbf{T}} \tag{7}$$

$$\mathbf{V}\mathbf{x} = \mathbf{H}\hat{\mathbf{q}} \tag{8}$$

In the above equations, the elements of $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{H}$ are given as

$$U_{IJ} = \int_{\Gamma_I} T_J^s v_I(Q) \mathrm{d}\Gamma \tag{9}$$

$$V_{IJ} = \int_{\Gamma_I} q_J^s v_I(Q) \mathrm{d}\Gamma \tag{10}$$

$$H_{IJ} = \int_{\Gamma_I} \Phi_J(\mathbf{s}) v_I(Q) \mathrm{d}\Gamma \tag{11}$$

where $v_I(Q)$ 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.

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

$$\mathbf{A}\mathbf{x} = \mathbf{d} \tag{12}$$

For the node $\mathbf{s}_I$, if $\tilde{T}$ is known, select the correspond row in $\mathbf{U}$ to $\mathbf{A}$, otherwise, select the correspond row in $\mathbf{V}$ to $\mathbf{A}$, and the corresponding term of $\mathbf{d}$ comes from the matrix–vector product of $\mathbf{H}\hat{\mathbf{T}}$ or $\mathbf{H}\hat{\mathbf{q}}$. Then the unknown vector $\mathbf{x}$ is obtained by solving the final algebraic equation. The nodal values $\hat{T}$ and $\hat{q}$ on the boundary can be computed by the back-substitution of $\mathbf{x}$ into Eqs. (7) and (8), then use Eqs. (1) and (2) the temperature field and normal flux on the boundary can be obtained.

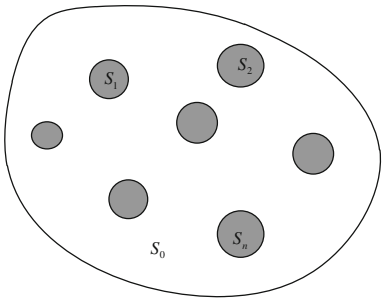


Fig. 1 The model of matrix with inclusions

3 New formulation of Hybrid BNM for composites with inclusions

For composite materials, a multi-domain formulation is a natural way to be chosen. In the conventional multi-domain solver [16], the unknown vectors of both the matrix domain and inclusion domain are computed in the final system equation. In this section, a new formulation for composites with inclusions is derived, which can reduce the total number of DOFs of the final system equation since only the vector of the matrix domain is computed.

Consider a matrix with inclusions as shown in Fig. 1, S_0 is the sub-domain of matrix and $S_1, S_2, \ldots, S_n$ are sub-domains of the inclusions. In this paper, we only consider the case when the inclusions are solid and totally embed in the matrix. For the sub-domain of the matrix, the following set of Hybrid BNM equation can be written as

$$\begin{bmatrix} \mathbf{U}_{00}^0 & \mathbf{U}_{01}^0 & \cdots & \mathbf{U}_{0n}^0 \\ \mathbf{U}_{10}^0 & \mathbf{U}_{11}^0 & \cdots & \mathbf{U}_{1n}^0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{U}_{n0}^0 & \mathbf{U}_{n1}^0 & \cdots & \mathbf{U}_{nn}^0 \end{bmatrix} \begin{Bmatrix} \mathbf{x}_0^0 \\ \mathbf{x}_1^0 \\ \vdots \\ \mathbf{x}_n^0 \end{Bmatrix} = \begin{Bmatrix} \mathbf{H}_0^0 \hat{\mathbf{T}}_0^0 \\ \mathbf{H}_1^0 \hat{\mathbf{T}}_1^0 \\ \vdots \\ \mathbf{H}_n^0 \hat{\mathbf{T}}_n^0 \end{Bmatrix} \tag{13}$$

$$\begin{bmatrix} \mathbf{V}_{00}^0 & \mathbf{V}_{01}^0 & \cdots & \mathbf{V}_{0n}^0 \\ \mathbf{V}_{10}^0 & \mathbf{V}_{11}^0 & \cdots & \mathbf{V}_{1n}^0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{V}_{n0}^0 & \mathbf{V}_{n1}^0 & \cdots & \mathbf{V}_{nn}^0 \end{bmatrix} \begin{Bmatrix} \mathbf{x}_0^0 \\ \mathbf{x}_1^0 \\ \vdots \\ \mathbf{x}_n^0 \end{Bmatrix} = \begin{Bmatrix} \mathbf{H}_0^0 \hat{\mathbf{q}}_0^0 \\ \mathbf{H}_1^0 \hat{\mathbf{q}}_1^0 \\ \vdots \\ \mathbf{H}_n^0 \hat{\mathbf{q}}_n^0 \end{Bmatrix} \tag{14}$$

where superscripts 0, subscripts 0 and k, $k = 1, \ldots, n$ stand for matrix, quantities exclusively associated with a domain, and quantities associated with the interface between the k-th inclusion and the matrix, respectively.

For the k-th inclusion domain we have

$$\mathbf{U}_{00}^k \mathbf{x}_0^k = \mathbf{H}_0^k \hat{\mathbf{T}}_0^k \tag{15}$$

$$\mathbf{V}_{00}^k \mathbf{x}_0^k = \mathbf{H}_0^k \hat{\mathbf{q}}_0^k \tag{16}$$

where superscripts k stands for the k-th inclusion and subscripts 0 indicates the quantities associated with the interface between the k-th inclusion and the matrix.

At the interface between a inclusion and the matrix, both the temperature and heat fluxes must be continuous. And if we use the same set of nodes distributed on the interface, we can obtain

$$\mathbf{H}_0^k \hat{\mathbf{T}}_0^k = \mathbf{H}_k^0 \hat{\mathbf{T}}_k^0 \tag{17}$$

$$\mathbf{H}_0^k \hat{\mathbf{q}}_0^k = -\mathbf{H}_k^0 \hat{\mathbf{q}}_k^0 \tag{18}$$

If use the conventional multi-domain, one can obtain the final system equation for the whole domain by using Eqs. (17) and (18) and the boundary conditions as

$$\begin{bmatrix} \mathbf{A}_{00}^0 & \mathbf{A}_{01}^0 & \cdots & \mathbf{A}_{0n}^0 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{U}_{10}^0 & \mathbf{U}_{11}^0 & \cdots & \mathbf{U}_{1n}^0 & -\mathbf{U}_{00}^1 & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{U}_{n0}^0 & \mathbf{U}_{n1}^0 & \cdots & \mathbf{U}_{nn}^0 & \mathbf{0} & \mathbf{0} & -\mathbf{U}_{00}^n \\ \mathbf{V}_{10}^0 & \mathbf{V}_{11}^0 & \cdots & \mathbf{V}_{1n}^0 & \mathbf{V}_{00}^1 & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{V}_{n0}^0 & \mathbf{V}_{n1}^0 & \cdots & \mathbf{V}_{nn}^0 & \mathbf{0} & \mathbf{0} & \mathbf{V}_{00}^n \end{bmatrix} \begin{Bmatrix} \mathbf{x}_0^0 \\ \mathbf{x}_1^0 \\ \vdots \\ \mathbf{x}_n^0 \\ \mathbf{x}_0^1 \\ \vdots \\ \mathbf{x}_0^n \end{Bmatrix} = \begin{Bmatrix} \mathbf{d}_0^0 \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{Bmatrix} \tag{19}$$

where $\mathbf{A}$ and $\mathbf{d}$ are given in the Sect. 2 according to the boundary condition of the matrix. In Eq. (19) all the unknown parameters $\mathbf{x}$ about the matrix and inclusions are solved.

From Eq. (13), one can obtain

$$\mathbf{H}_k^0 \hat{\mathbf{T}}_k^0 = \mathbf{U}_{k0}^0 \mathbf{x}_0^0 + \mathbf{U}_{k1}^0 \mathbf{x}_1^0 + \cdots + \mathbf{U}_{kn}^0 \mathbf{x}_n^0 = \sum_{j=0}^{n} \mathbf{U}_{kj}^0 \mathbf{x}_j^0 \tag{20}$$

From Eq. (15), one can obtain

$$\mathbf{x}_0^k = [\mathbf{U}_{00}^k]^{-1} \mathbf{H}_0^k \hat{\mathbf{T}}_0^k \tag{21}$$

Using Eqs. (21), (17) and (20), one can obtain

$$\mathbf{x}_0^k = [\mathbf{U}_{00}^k]^{-1} \mathbf{H}_k^0 \hat{\mathbf{T}}_k^0 = [\mathbf{U}_{00}^k]^{-1} \sum_{j=0}^{n} \mathbf{U}_{kj}^0 \mathbf{x}_j^0 \tag{22}$$

Substitute Eq. (22) into Eq. (16), and use Eq. (18), we can obtain

$$-\mathbf{H}_k^0 \hat{\mathbf{q}}_k^0 = \mathbf{H}_0^k \hat{\mathbf{q}}_0^k = \mathbf{V}_{00}^k \mathbf{x}_0^k = \mathbf{V}_{00}^k [\mathbf{U}_{00}^k]^{-1} \sum_{j=0}^{n} \mathbf{U}_{kj}^0 \mathbf{x}_j^0 \tag{23}$$

Then by substituting Eq. (23) into Eq. (14) and assembling with Eq. (13) by the boundary condition, we can obtain the final system equations as

$$\begin{bmatrix} \mathbf{A}_{00}^0 & \mathbf{A}_{01}^0 & \cdots & \mathbf{A}_{0n}^0 \\ \mathbf{D}_{10}^0 & \mathbf{D}_{11}^0 & \cdots & \mathbf{D}_{1n}^0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{D}_{n0}^0 & \mathbf{D}_{n1}^0 & \cdots & \mathbf{D}_{nn}^0 \end{bmatrix} \begin{Bmatrix} \mathbf{x}_0^0 \\ \mathbf{x}_1^0 \\ \vdots \\ \mathbf{x}_n^0 \end{Bmatrix} = \begin{Bmatrix} \mathbf{d}_0^0 \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{Bmatrix} \tag{24}$$

where

$$\mathbf{D}_{kj}^0 = \mathbf{V}_{kj}^0 + \mathbf{C}^k \mathbf{U}_{kj}^0 \tag{25}$$

$$\mathbf{C}^k = \mathbf{V}_{00}^k [\mathbf{U}_{00}^k]^{-1} \tag{26}$$

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 the similar relationship between inclusions phases of different size and material, which can be expressed as below

$$\mathbf{C}^{k+1} = \frac{\kappa^{k+1}}{\kappa^k} \frac{r^k}{r^{k+1}} \mathbf{C}^k \tag{27}$$

where κ^k and r^k are heat conductivity and radius of the k-th inclusion, respectively.

By solving Eq. (24), the unknown vector $\mathbf{x}$ for the matrix domain can be obtained. Then the other unknown vector $\mathbf{x}$ for the inclusion domains can be computed by Eq. (22). Equation (24) is a new formulation for the analysis of composites with inclusions, which can reduce the total DOFs comparing with Eq. (19) since only the unknown vector $\mathbf{x}$ for the matrix domain are assembled in the final system equation.

4 Formulation of new fast multipole method for hybrid boundary node method

The coefficient matrix in Eq. (24) is unsymmetrical and dense, which makes it is impossible to apply the method for large scale computation. In this paper, an iterative solver, i.e. the restarted preconditioned GMRES [34] is used to solve Eq. (24). The most time-consuming aspect of an iterative method when employed for solving a system of linear equations is the matrix–vector product in each iteration.

Equation (24) can be rewritten as

$$\begin{bmatrix} \mathbf{A}_{00}^0 & \mathbf{A}_{01}^0 & \cdots & \mathbf{A}_{0n}^0 \\ \mathbf{V}_{10}^0 & \mathbf{V}_{11}^0 & \cdots & \mathbf{V}_{1n}^0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{V}_{n0}^0 & \mathbf{V}_{n1}^0 & \cdots & \mathbf{V}_{nn}^0 \end{bmatrix} \begin{Bmatrix} \mathbf{x}_0^0 \\ \mathbf{x}_1^0 \\ \vdots \\ \mathbf{x}_n^0 \end{Bmatrix} + \begin{bmatrix} \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{U}_{10}^0 & \mathbf{U}_{11}^0 & \cdots & \mathbf{U}_{1n}^0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{U}_{n0}^0 & \mathbf{U}_{n1}^0 & \cdots & \mathbf{U}_{nn}^0 \end{bmatrix} \begin{Bmatrix} \mathbf{x}_0^0 \\ \mathbf{x}_1^0 \\ \vdots \\ \mathbf{x}_n^0 \end{Bmatrix} \circ \begin{bmatrix} \mathbf{0} \\ \mathbf{C}^1 \\ \vdots \\ \mathbf{C}^n \end{bmatrix} = \begin{Bmatrix} \mathbf{d}_0^0 \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{Bmatrix} \tag{28}$$

where $\circ$ is Hadamard product. In Eq. (28), $\mathbf{C}^k$ is relative to the k-th inclusion and its dimension is small sine we can discrete every inclusion with not many nodes, and $\mathbf{C}^k$ can be computed directly. Consider an iteration vector $\mathbf{x}'$ corresponding to the solution vector $\mathbf{x}$, then the most time-consuming aspect of the matrix–vector product in Eq. (28) is one of the following sums:

$$\sum_{J=1}^{N} \int_{\Gamma_I} T_J^s v_I(Q) x'_J \, \mathrm{d}\Gamma \tag{29}$$

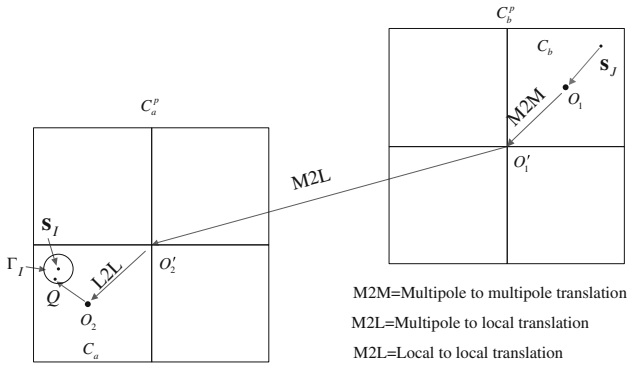


Fig. 2 Conversions of multipole to local expansions

$$\sum_{J=1}^{N}\int_{\Gamma_I} q_J^s v_I(Q)x_J'\,\mathrm{d}\Gamma \tag{30}$$

where x_J' is the Jth element of the iteration vector $\mathbf{x}'$.

The sums of (29) and (30) will be accelerated by the FMM. The main idea of the FMM is to translate the node-to-node interactions to cell-to-cell interactions. In 3D problems, the cells can be constructed by a hierarchical oct-tree structure.

The sum of (29) will be computed for convenience and sum of (30) can be computed in the same way. Instead of treating interactions with each of the distant nodes individually, the FMM computes cell–cell interactions and achieves an $O(N)$ complexity algorithm.

4.1 Formulation for the original FM-HBNM

We first introduce the original FM-HBNM [21] briefly before the formulation of the new FM-HBNM is derived. This is because the new FM-HBNM is presented as a modification to the original one. Now consider two cells C_a and C_b, which contain N_a and N_b nodes, respectively. The fundamental solution in Eq. (5) is expanded in terms of solid harmonic series as

$$\begin{aligned} T_J^s &= \frac{1}{4\pi\kappa}\frac{1}{r(Q,\mathbf{s}_J)} \\ &= \frac{1}{4\pi\kappa}\sum_{n=0}^{\infty}\sum_{m=-n}^{n} R_{n,m}(\overrightarrow{O_1 s_J})\overline{S_{n,m}}(\overrightarrow{O_1 Q}) \end{aligned} \tag{31}$$

for $|\overrightarrow{O_1 Q}| > |\overrightarrow{O_1 s_J}|$, where $R_{n,m}$ and $S_{n,m}$ are functions defined in [30], and a superposed bar indicates the complex conjugate. O_1 is the center of C_b (see Fig. 2).

Substituting Eq. (31) into sum of (29) and with the summation over the nodes included in C_b, we can obtain

$$\begin{aligned} &\sum_{J=1}^{N_b}\int_{\Gamma_I} T_J^s v_I(Q)x_J'\,\mathrm{d}\Gamma(Q) \\ &= \sum_{n=0}^{\infty}\sum_{m=-n}^{n}\frac{1}{4\pi\kappa}\int_{\Gamma_I}\overline{S_{n,m}}(\overrightarrow{O_1 Q})v_I(Q)\mathrm{d}\Gamma(Q)M_{n,m}(O_1) \end{aligned} \tag{32}$$

where $M_{n,m}(O_1)$ is multipole moment centered at O_1, expressed as

$$M_{n,m}(O_1) = R_{n,m}(\overrightarrow{O_1 s_J})x_J' \tag{33}$$

Suppose that C_a and C_b are belong to two larger cells C_a^p and C_b^p, known as the parent cells of C_a and C_b, respectively. Assume that C_a^p and C_b^p are still far away each other (see Fig. 2).

The multipole moment about the center of C_b to the center of C_b^p can be translated by the following equation

$$\begin{aligned} &M_{n,m}(O_1') = \\ &\sum_{n'=0}^{n}\sum_{m'=-n'}^{n'} R_{n',m'}(\overrightarrow{O_1' O_1})M_{n-n',m-m'}(O_1) \end{aligned} \tag{34}$$

which is called as the multipole to multipole (M2M) translation.

Since C_a^p and C_b^p are still far away each other, then the multipole moment from O_1' to O_2' can be translated by

$$\begin{aligned} &L_{n',m'}(O_2') = \\ &\sum_{n=0}^{\infty}\sum_{m=-n}^{n}(-1)^{n'}\overline{S_{n+n',m+m'}}(\overrightarrow{O_1' O_2'})M_{n,m}(O_1') \end{aligned} \tag{35}$$

The above equation is called as M2L translation.

At last, the center of the local moment will be shift from the center of C_a^p to the center of C_a by the following equation

$$\begin{aligned} &L_{n',m'}(O_2) = \\ &\sum_{n=n'}^{\infty}\sum_{m=-n}^{n} R_{n-n',m-m'}(\overrightarrow{O_2' O_2})L_{n,m}(O_2') \end{aligned} \tag{36}$$

which is called as local to local (L2L) translation and O_2 is the center of C_a.

Then Eq. (32) can be rewritten as

$$\begin{aligned} &\sum_{J=1}^{N_b}\int_{\Gamma_I} T_J^s v_I(Q)x_J'\,\mathrm{d}\Gamma(Q) = \frac{1}{4\pi\kappa} \\ &\times\sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'}\int_{\Gamma_I} R_{n',m'}(\overrightarrow{O_2 Q})v_I(Q)\mathrm{d}\Gamma(Q)L_{n',m'}(O_2) \end{aligned} \tag{37}$$

The computation of sum (30) is the same as that of sum (29), except that Eq. (37) is replaced by

$$\sum_{J=1}^{N_b}\int_{\Gamma_I} q_J^s v_I(Q)x'_J \mathrm{d}\Gamma(Q) = -\frac{1}{4\pi}\sum_{n'=0}^{\infty}\sum_{m'=-n'}^{n'} \int_{\Gamma_I} \frac{\partial R_{n',m'}(\overrightarrow{O_2Q})}{\partial n} v_I(Q)\mathrm{d}\Gamma(Q) L_{n',m'}(O_2) \tag{38}$$

In the procedure of the FM-HBNM, a hierarchical decomposition of space, represented by a tree structure is employed. Consider a smallest cube that can contain the entire domain as the root cell at level 0. Then divide this root cube into eight equal smaller cubes, which can be called as cubes of level 1. Continue dividing in this way, that is, the cubes of level $l+1$ is obtained from level l by subdividing of a cube into eight equal cubes. The eight cubes at level $l+1$ obtained by subdivision of the cube at level l are considered as its children. Stop the subdivision of a cube while the number of nodes included in the cube is smaller than a prescribed number. Delete the child cube if it contains no node. A childless cube we call it leaf. Two cubes are said to be neighbors if they are at the same level and share at least a vertex (a cube is a neighbor of itself). Two cubes are said to be well separated if they are at the same level but not neighbors. Each cube b has an interaction list, whose members are the children of the neighbors of b's parent which are well separated from cube b.

The multipole and local moments are orchestrated in the tree structure in a recursive way, which consists of two basic steps: the *upward pass* and the *downward pass*. During the upward pass, the multipole moments are accumulated from leaves to the root of the tree by Eq. (34). During the downward pass, local moments are distributed form the root to the leaves by Eqs. (35) and (36).

In the algorithm of FM-HBNM introduced above, if one truncates the infinite series in the multipole expansion taking p terms then the computational costs for M2M, M2L and L2L translations are proportional to $O(p^4)$, $O(189p^4)$ in the worst case and $O(p^4)$, respectively. The M2L translation is the bottleneck in the algorithm of FMM and its computational cost relate with the number of boxes in the interaction list. In order to reduce the cost for M2L, Greengard and Rokhlin have developed a new version of FMM [29], in which they use a new diagonal form for translation operators and further reduce the computational cost of M2L translation to $O(p^3)$. In the following section, the formulation for the new version of FM-HBNM will be derived.

4.2 Formulation for the new FM-HBNM

In the new FMM, three translations are introduced to replace the M2L translation, which called multipole moment to exponential expansion (M2X) translation, exponential expansion to exponential expansion (X2X) translation and exponential expansion to local expansion (X2L) expansion (see Fig. 3). The computational costs for M2X, X2X, X2L translations are $O(p^3)$, $O(p^2)$ and $O(p^3)$, respectively. This is why the new FMM is more efficient than the original one.

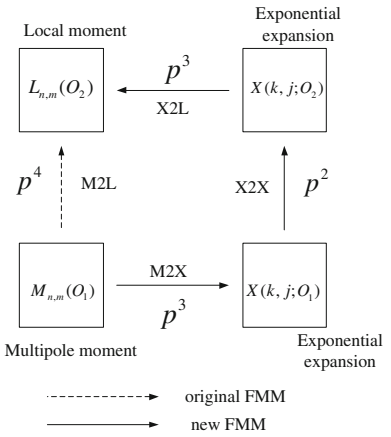


Fig. 3 Computational cost for M2L, and M2X+X2X+X2L

Suppose we evaluate the local expansion associated with a cell B from another cell C, which is belonging to the interaction list of B. We divided the interaction list of a box B into 6 lists called the *up-list*, *down-list*, *north-list*, *south-list*, *east-list* and *west-list*. The up-list and down-list contain boxes located in the $+z$ and $-z$ directions of B respectively. The north-list and south-list contain boxes located in the $+y$ and $-y$ directions of B except those in the up-list or down-list, respectively. The east-list and west-list contain boxes in the $+x$ and $-x$ directions of B except that those in the up-list, down-list, north-list or south-list, respectively.

Now suppose the cell C is in the down-list of cell B (see Fig. 4). Give two points O at (x_0, y_0, z_0) and Q at (x, y, z), with Q in the box B and O in box C. Since $z > z_0$ holds, we have the well-known integral representation

$$\frac{1}{\sqrt{(x-x_0)^2+(y-y_0)^2+(z-z_0)^2}} = \frac{1}{2\pi}\int_0^{\infty} e^{-\lambda(z-z_0)}\int_0^{2\pi} e^{i\lambda((x-x_0)\cos\alpha+(y-y_0)\sin\alpha)}\mathrm{d}\alpha\mathrm{d}\lambda \tag{39}$$

The outer integral with respect to λ in Eq. (39) is computed with generalized Gaussian quadrature while the inner integral with respect to α is computed with the trapezoidal rule, then

$$\frac{1}{\sqrt{(x-x_0)^2+(y-y_0)^2+(z-z_0)^2}} = \sum_{k=1}^{s(\varepsilon)}\sum_{j=1}^{M(k)} \frac{w_k}{M(k)d} e^{-(\lambda_k/d)(z-z_0)} e^{i(\lambda_k/d)((x-x_0)\cos\alpha_j(k)+(y-y_0)\sin\alpha_j(k))} + \varepsilon \tag{40}$$

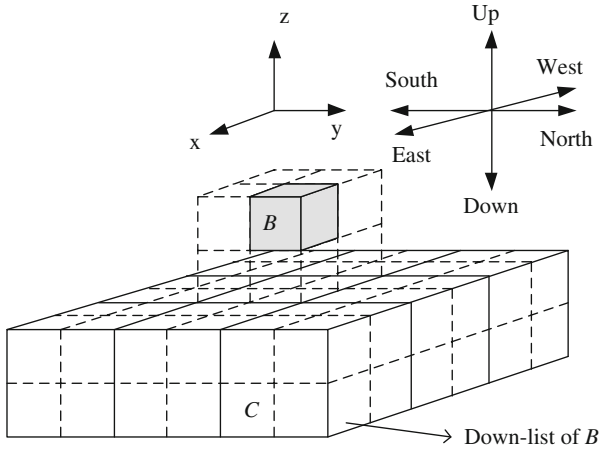


Fig. 4 Down-list of box B

where $\alpha_j(k)$ is given by

$$\alpha_j(k) = \frac{2\pi j}{M(k)}$$

ε is the error term and the numbers $s(\varepsilon)$, $M(k)$, Gaussian weights w_k and nodes λ_k are given in [29,35]. d is the length of box B.

Noting we have the following formulae

$$S_{n,m}(\overrightarrow{Ox}) = (-1)^n \partial_{x+}^m \partial_{x_3}^{n-m} \left(\frac{1}{|\overrightarrow{Ox}|} \right) \quad (m \geq 0) \tag{41}$$

$$S_{n,-m}(\overrightarrow{Ox}) = (-1)^m \overline{S_{n,m}}(\overrightarrow{Ox}) = (-1)^{n+m} \partial_{x-}^m \partial_{x_3}^{n-m} \left(\frac{1}{|\overrightarrow{Ox}|} \right) \quad (m \geq 0) \tag{42}$$

$$\partial_{x\pm} = \left(\frac{\partial}{\partial x_1} \pm i \frac{\partial}{\partial x_2} \right) \tag{43}$$

Assume O_1 is the center of cell C and O_2 is the center of cell B. Now we can rewrite Eq. (32) by using Eqs. (41) and (42) as

$$\sum_{J=1}^{N_b} \int_{\Gamma_I} T_J^s v_I(Q) x'_J \mathrm{d}\Gamma(Q) = \frac{1}{4\pi\kappa} \sum_{k=1}^{s} \sum_{j=1}^{M(k)} X(k, j; O_1) \int_{\Gamma_I} [e^{-(\lambda_k/d)((\overrightarrow{O_1Q})_3 - i(\overrightarrow{O_1Q})_1 \cos\alpha_j(k) - i(\overrightarrow{O_1Q})_2 \sin\alpha_j(k))} \times v_I(Q)] \mathrm{d}\Gamma(Q) \tag{44}$$

where $X(k, j; O_1)$ is the coefficient of the exponential expansion centered at O_1 defined as

$$X(k, j; O_1) = \frac{w_k}{M(k)d} \sum_{m=-\infty}^{\infty} (-i)^m e^{-im\alpha_j(k)} \times \sum_{n=|m|}^{\infty} (\lambda_k/d)^n M_{n,m}(O_1) \tag{45}$$

Equation (45) converts the multipole moments into the exponential expansion coefficients and we call it M2X translation.

Then we shift the center of the exponential expansion from O_1 to O_2 and we can obtain

$$\sum_{J=1}^{N_b} \int_{\Gamma_I} T_J^s v_I(Q) x'_J \mathrm{d}\Gamma(Q) = \frac{1}{4\pi\kappa} \sum_{k=1}^{s} \sum_{j=1}^{M(k)} X(k, j; O_2) \int_{\Gamma_I} [e^{-(\lambda_k/d)((\overrightarrow{O_2Q})_3 - i(\overrightarrow{O_2Q})_1 \cos\alpha_j(k) - i(\overrightarrow{O_2Q})_2 \sin\alpha_j(k))} \times v_I(Q)] \mathrm{d}\Gamma(Q) \tag{46}$$

where

$$X(k, j; O_2) = X(k, j; O_1) e^{-(\lambda_k/d)((\overrightarrow{O_1O_2})_3 - i(\overrightarrow{O_1O_2})_1 \cos\alpha_j(k) - i(\overrightarrow{O_1O_2})_2 \sin\alpha_j(k))} \tag{47}$$

We call Eq. (47) as the X2X translation.

Substituting the following formula into the right-hand side of Eq. (46):

$$e^{-(\lambda_k/d)((\overrightarrow{O_2Q})_3 - i(\overrightarrow{O_2Q})_1 \cos\alpha_j(k) - i(\overrightarrow{O_2Q})_2 \sin\alpha_j(k))} = \sum_{n=0}^{\infty} \frac{(-\lambda_k/d)^n}{n!} \times ((\overrightarrow{O_2Q})_3 - i(\overrightarrow{O_2Q})_1 \cos\alpha_j(k) - i(\overrightarrow{O_2Q})_2 \sin\alpha_j(k))^n \tag{48}$$

and use the following identity

$$\frac{((\overrightarrow{Ox})_3 - i(\overrightarrow{Ox})_1 \cos\alpha_j(k) - i(\overrightarrow{Ox})_2 \sin\alpha_j(k))^n}{n!} = \sum_{m=-n}^{n} (-i)^m e^{-im\alpha_j(k)} R_{n,m}(\overrightarrow{Ox}) \tag{49}$$

one can rewrite Eq. (46) as

$$\sum_{J=1}^{N_b} \int_{\Gamma_I} T_J^s v_I(Q) x'_J \mathrm{d}\Gamma(Q) = \frac{1}{4\pi\kappa} \times \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \int_{\Gamma_I} R_{n,m}(\overrightarrow{O_2Q}) v_I(Q) \mathrm{d}\Gamma(Q) L_{n,m}(O_2) \tag{50}$$

where

$$L_{n,m}(O_2) = \sum_{k=1}^{s} (-\lambda_k/d)^n \sum_{j=1}^{M(k)} (-i)^m e^{-im\alpha_j(k)} X(k, j; O_2) \tag{51}$$

Equation (51) is called as X2L translation.

The discussion in the previous section has been restricted to the case where the box C is in the down-list of B. Now we consider the general case. We rotate the coordinate system so

that the cell B is in the positive $\widetilde{z}$ direction viewed from the cell C if the cell C is included in lists other that the down-list, where the $(\widetilde{Ox})_i$ denotes the new axis. The explicit form of the coefficient of rotation is given by

$$\Re_{n,m,m'}(\nu,\alpha) = (-1)^{m+m'}(n+m')!(n-m')! \sum_k \frac{(\alpha_0 - i\alpha_3)^{n+m-k}(-i\alpha_1-\alpha_2)^{m'-m+k}(-i\alpha_1+\alpha_2)^k(\alpha_0+i\alpha_3)^{n-m'-k}}{(n+m-k)!(m'-m+k)!k!(n-m'-k)!} \tag{52}$$

where ν is a unit vector parallel to the rotation axis and α is a rotation angle. $\alpha_0 = \cos(\alpha/2)$ and $\alpha_i = \nu_i \sin(\alpha/2)$. The summation is over such k that the numbers in the parentheses in the denominator are all non-negative.

The M2L translation process for the case when $C \in$ lists other than the down-list can be described as below.

Rotation

Rotate the multipole moments by the following equation

$$M^*_{n,m}(O_1) = \sum_{m'=-n}^{n} \Re_{n,m,m'}(\nu,\alpha) M_{n,m'}(O_1) \tag{53}$$

Compute the coefficients of the exponential expansion

Compute the coefficients of the exponential expansion as follows:

$$X^*(k,j;O_1) = \frac{w_k}{M(k)d} \sum_{m=-\infty}^{\infty} (-i)^m e^{-im\alpha_j(k)} \sum_{n=|m|}^{\infty} (\lambda_k/d)^n M^*_{n,m}(O_1) \tag{54}$$

Translate the coefficients of the exponential expansion

As the center of the exponential expansion is shifted from O_1 (the centroid of box C) to O_2 (the centroid of box B), the coefficients of the exponential expansion are transformed by the following equation

$$X^*(k,j;O_2) = X^*(k,j;O_1) \times e^{-(\lambda_k/d)((\widetilde{O_1O_2})_3 - i(\widetilde{O_1O_2})_1 \cos\alpha_j(k) - i(\widetilde{O_1O_2})_2 \sin\alpha_j(k))} \tag{55}$$

where the components $(\widetilde{O_1O_2})_i$ are obtained by applying the corresponding rotation of coordinates to $(\overrightarrow{O_1O_2})$ and $(\widetilde{O_1O_2})_i = A_{ij}(\overrightarrow{O_1O_2})_j$, and

$$\mathbf{A}^D = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{A}^U = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

$$\mathbf{A}^S = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{A}^N = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$$

$$\mathbf{A}^W = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad \mathbf{A}^E = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$$

and $\mathbf{A}^D, \mathbf{A}^U, \mathbf{A}^S, \mathbf{A}^N, \mathbf{A}^W, \mathbf{A}^E$ indicate the rotation matrix of the down-list, up-list, south-list, north-list, west-list and east-list, respectively.

Compute the coefficients of the local expansion

Compute the coefficients of the local expansion via Eq. (51) as follows:

$$L^*_{n,m}(O_2) = \sum_{k=1}^{s} (-\lambda_k/d)^n \sum_{j=1}^{M(k)} (-i)^m e^{-im\alpha_j(k)} X^*(k,j;O_2) \tag{56}$$

Rotation

At last, rotate the local expansion to the original coordinate system.

$$L_{n,m}(O_2) = \sum_{m'=-n}^{n} \Re_{n,m',m}(\nu,\alpha) L^*_{n,m'}(O_2) \tag{57}$$

In Eqs. (53) and (57), the vector ν and rotation angle α are given as

If $C \in$ up-list, $\nu = e_1, \alpha = \pi$;
If $C \in$ north-list, $\nu = e_1, \alpha = -\pi/2$;
If $C \in$ south-list, $\nu = e_1, \alpha = \pi/2$;
If $C \in$ east-list, $\nu = e_2, \alpha = \pi/2$;
If $C \in$ west-list, $\nu = e_2, \alpha = -\pi/2$.
e_i is the base vector for the Cartesian coordinates.

Finally the coefficients of local expansion at O_2 can be obtained by add the local expansion computed from the up-list, down-list, north-list, south-list, east-list and west-list of B.

4.3 Algorithm of the new FM-HBNM

The algorithm of the new FM-HBNM can be summarized as below:

Step 1: *Discretization* Create nodes which disturbed on the boundary of the domain in the same manner as in the original Hybrid BNM.

Step 2: *Construction of oct-tree structure* Construct the hierarchy of boxes using an oct-tree data structure described in the original FM-HBNM.

Step 3: *Computation of the multipole moments (upward)* For an iteration vector x'_J form multipole moments $M_{n,m}$ about the center of each leaf from all the nodes included in that leaf by Eq. (33). Now consider a non leaf box of level l. Compute multipole moment $M_{n,m}$ about the centre of each box at level l by merging multipole moments from its children using Eq. (34) (M2M in Fig. 2). This procedure is repeated for $l \geq 2$ tracing the tree structure of boxes obtained in step upward (decreasing l).

Step 4: *Compute the exponential expansion* Rotating the multipole moments in each box via Eq. (53), then compute the exponential expansion by Eq. (54).

Step 5: *Compute the local expansion* Consider boxes at level l from level 2 to the finest level. For each box B at level l, we first use Eq. (55) to shift the center of the exponential expansion for each box C (which in the interaction list of box B) to box B, depending on the position of C relative to B. Next convert the exponential expansions from all the up-list, down-list, north-list, south-list, east-list and west-list to the local expansion via Eq. (56) and rotate them via Eq. (57). Then add them together to obtain the local expansion of box B. If $l > 2$, shift the local expansion of B's parent to itself, using Eq. (36) (L2L in Fig. 2).

Add these two local expansions together.

Step 6: *Evaluation of the integral in sum* (29) *and* (30): The integral in sums (29) and (30) are computed by Eqs. (37) and (38).

Step 7: *Update* Update the candidate vector and go back to step 3.

A good preconditioner is crucial for the convergence when use the GMRES solver. In this paper, we use the preconditioner obtained by inverting the diagonal blocked submatrices in Eq. (24). The inverse of $\mathbf{A}_{00}^0$ is approximated by the conventional single domain preconditioner, which is based on the leaves of the oct-tree, and the inverse of $\mathbf{D}_{kk}^0$ is obtained according to each inclusion.

5 Numerical results

The proposed techniques have been implemented in C++. In this section, three numerical examples are presented to demonstrate the performance of the method. In the new FM-HBNM, a restarted preconditioned GMRES(m) with m=25 is employed as the iterative solver. The infinite expansions in the new FM-HNBM are truncated after $p = 10$ and the exponential expansion order $s = 8$ given in [35]. The maximum number of boundary nodes in a leaf box is set to be 60 and the iteration is terminated when the relative error is less than 10^{-5}. All the computations are performed on a PC with a 3.4 GHz CPU and 16.0 GB RAM.

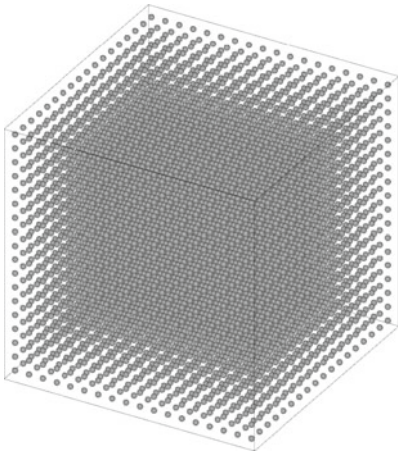


Fig. 5 A cube with 5832 inclusions

For the purpose of error estimation, a formula is defined as

$$error = \frac{1}{|u|_{max}} \sqrt{\frac{1}{N} \sum_{i=1}^{N} (u_i^{(e)} - u_i^{(n)})^2} \tag{58}$$

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.

5.1 Accuracy and efficiency of the presented method

In order to examine the accuracy and efficiency of the method described above, models of the cubes with increasing number of spherical inclusions are considered. Each model of the cube is bounded by the planes $x = \pm 1, y = \pm 1, z = \pm 1$, containing a total of $m \times m \times m$ inclusions, with $m = 2, 4, 6, 8, 10, 12, 14, 16$and 18. The radiuses of the holes are $0.4/m$ for the models contain $m \times m \times m$ holes. In all the models, the inclusions are distributed uniformly (see Fig. 5) and the coordinate systems are set in the center of the geometries.

The cubes and inclusions are assumed to be with the same materials. Potential problems with Dirichlet boundary condition on all the outer faces are considered, according to the following exact solution:

$$u = x^3 + y^3 + z^3 - 3yz^2 - 3xz^2 - 3zy^2 \tag{59}$$

The model with $m = 2$ (with 8 inclusions) is studied first and several different node distributions are compared to show the accuracy and convergence of the method. The relative error of u is computed by nodes on the line (−1,0,0) to (1,0,0) and the results are shown in Fig. 6. Both Gaussian elimination method and new FMM are applied to solve the problems and the Gaussian elimination method is applied when the DOFs less than 10,000. In the new FM-HBNM, we have used two different exponential expansion order s. From Fig. 6 we can find the method with direct solver has high

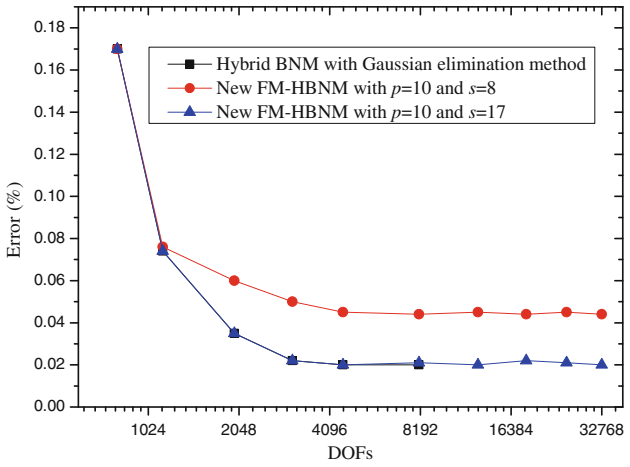


Fig. 6 Relative error

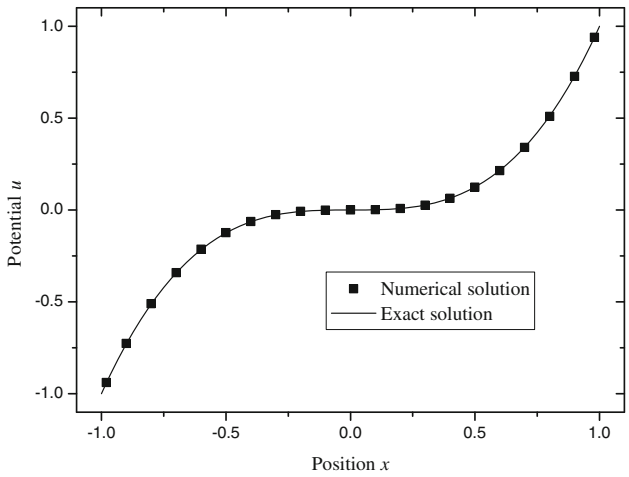


Fig. 7 Potential u along the x direction

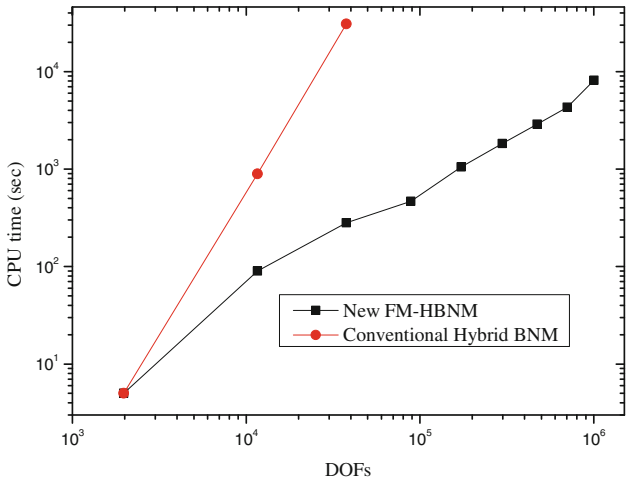


Fig. 8 CPU time for solving the final system equation

accuracy. The accuracy declines while the new FMM with $s = 8$ is applied, but it remains high. And if we use higher order $s = 17$, the accuracy of the new FM-HBNM increases. Actually, the precision of the FMM can be controlled [35] by choosing different order p of infinite expansions and different exponential expansion order s. Sine more time will be costed by using higher order s and the accuracy is high enough in our method with order $s = 8$, thus we use $s = 8$ to ensure the efficiency of the new FM-HBNM.

Then we consider models with m increasing from 2 to 18. In each model, 600 nodes are distributed on the outer faces of the cubes and 172 nodes are distributed on each interface between the matrix and inclusions. The maximum number of total DOFs in the final system equation is 1,003,704. However, if the conventional multi-domain formulation is applied, the maximum number of DOFs would reach to 2,006,808.

Figure 7 shows the value of potential u along the line $y = z = 0$ for the case with 5,832 inclusions and we can observe that the numerical result has good agreement with the exact solution. The CPU time for solving the final system equation is plotted in Fig. 8. In order to show the efficient of the new FM-HBNM, the CPU time of the conventional Hybrid BNM with the Gaussian elimination method is also plotted in Fig. 8. One can observe that the new FM-HBNM is much efficient than the conventional Hybrid BNM and it can solve more than 1,000,000 DOFs on a personal computer with reasonable time.

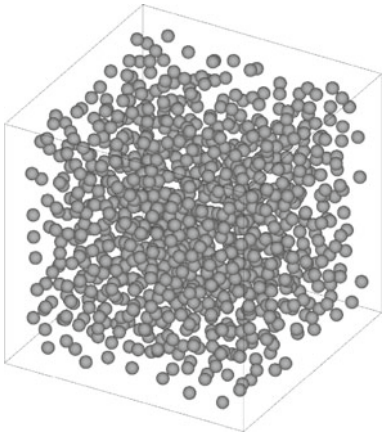
5.2 Thermal behavior of composites with spherical inclusions

In this example, the thermal behavior of cubes with 1,000 randomly distributed spherical inclusions (see Fig. 9) is studied. The parameters of the matrix are: the length $L = 200$ m, the conductivity $\kappa_{matrix} = 1$ W/m K. Uniform temperature of 100 K is imposed at the left surface of the cube while 0K is imposed at the right surfaces, respectively, and the other four surfaces are heat flux free. These boundary conditions allow us to estimate the effective heat conductivity of the mode in the axial direction. The formula for the effective heat conductivity can be written as

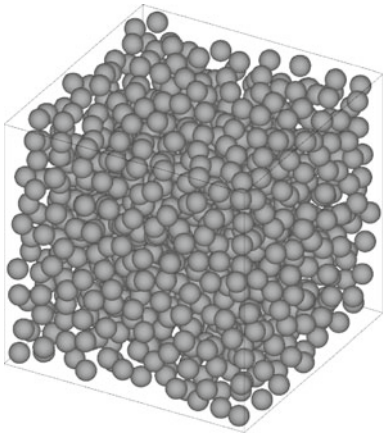
$$\kappa_e = -\frac{qL}{\Delta T} \tag{60}$$

where κ_e is the effective heat conductivity, q is the average value of normal flux at the two end face, L is the length of the cube, ΔT is the temperature difference between the two end face.

Two models are considered. In the first model, the parameters of the inclusions are: the radius $r = 5$m and the thermal conductivity $\kappa_{inclusion}$ changes from 1 to 10 W/m K. In the second model, the parameters of the inclusions are: the thermal conductivity $\kappa_{inclusion} = 6$ W/m K and the radius r changes from 2 to 8 m (see Fig. 9b). In all the models, 600



(a) 1000 inclusions (r=5m)



(b) 1000 inclusions (r=8m)

Fig. 9 Cubes with 1,000 inclusions

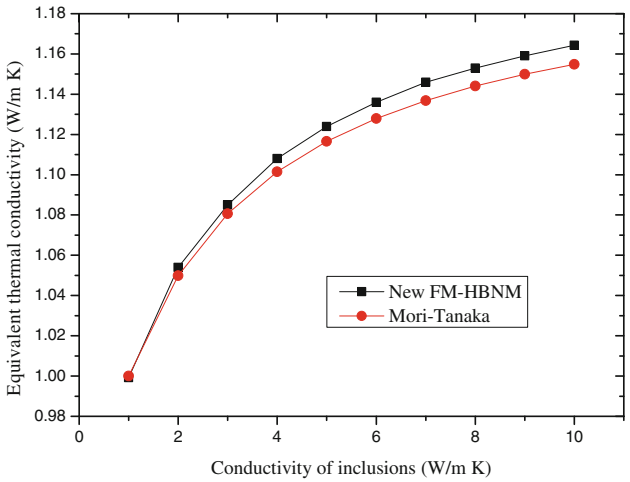


Fig. 10 Equivalent thermal conductivity with conductivity of inclusions

nodes distribute on the outer faces and 172 nods distribute on each interface.

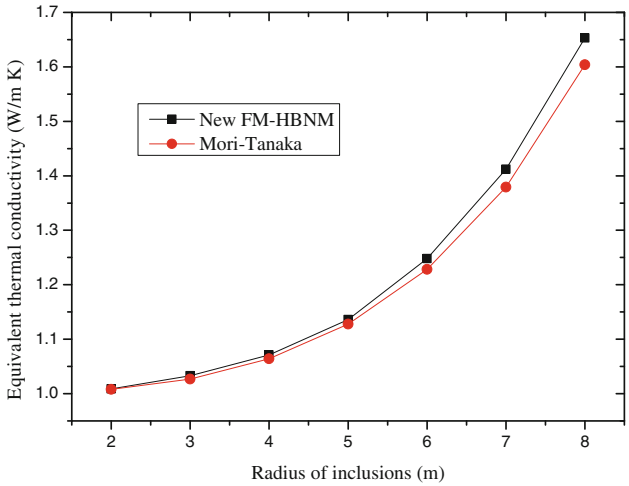


Fig. 11 Equivalent thermal conductivity with radius of inclusions

Figures 10 and 11 show the equivalent thermal conductivities for the first and second models, respectively. In order to show the accuracy of the new formulation, the Mori–Tanaka method [36] are also used. The main formulation of Mori–Tanaka method is

$$\kappa_e = \kappa_0 + \frac{\sum_{j=1}^{n}(\kappa_j - \kappa_0)c_j \bar{H}^j/\bar{H}^0}{c_0 + \sum_{i=1}^{n} c_i \bar{H}^i/\bar{H}^0} \tag{61}$$

where $\kappa_i, i = 0, 1, 2, \ldots, n$ and c_i are thermal conductivity and volume fraction for material i and $i = 0$ corresponds to the matrix material. For spherical particles

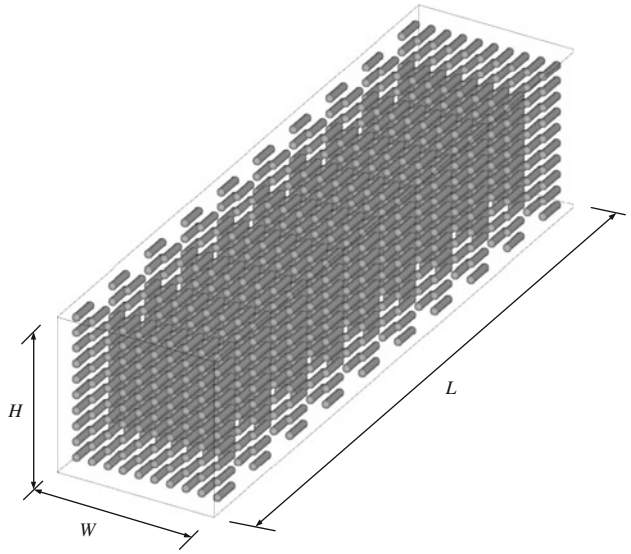
$$\bar{H}^j/\bar{H}^0 = \frac{3\kappa_0}{2\kappa_0 + \kappa_j} \tag{62}$$

From Figs. 10 and 11 we can observe that there is a little difference between our method and Mori–Tanaka method. One of the reasons is that Mori–Tanaka method is also not an exact solution.

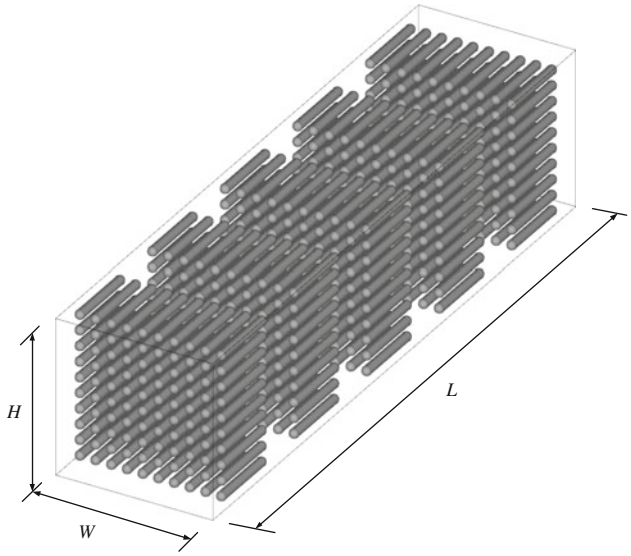
In this example, the total number of DOFs in the final system equation is about 172,600 and the CPU time for the conventional Hybrid BNM with Gaussian elimination method will be more than 40 days, however, about half an hour is taken by the new FM-HBNM presented.

5.3 Thermal behavior of fiber reinforced composites

This example studies the thermal behavior of fiber reinforced composites (see Fig. 12). In all the models, the length of the matrix is $L = 800$ m, the height of the matrix is H=200 m, the width of the matrix is $W = 200$ m (see Fig. 12) and the conductivity $\kappa_{matrix} = 1$ W/m K. Uniform temperature of 100 K is imposed at the left end while 0 K is imposed at the right end, respectively, and the other four surfaces are heat flux free. The geometry of the fiber is shown in Fig. 13. Five cases are studied and the fibers are uniformly distributed in



(a) 1000 shorter fibers



(b) 500 longer fibers

Fig. 12 Fiber reinforced composites

all the models. The parameters of the fiber inclusions can be seen in Table 1. In Table 1, the fourth column indicates the conductivity of the fiber inclusion and the fifth column lists the total number of inclusions for each case.

The fiber conductivity κ_{fiber}, radius r and length L_c on the equivalent thermal conductivities of the composites are studied in Cases A, B and C, respectively, and the results are shown in Figs. 14, 15 and 16, respectively. The results are also compared with Mori–Tanaka method and the fibers are treated as spheroids. For Mori–Tanaka method, Eq. (61) can also be used, and for prolate spheroid with $x_1 > x_2 = x_3$, where x_1, x_2, and x_3 are the semiaxes of the spheroid, it has the following equations [36,37].

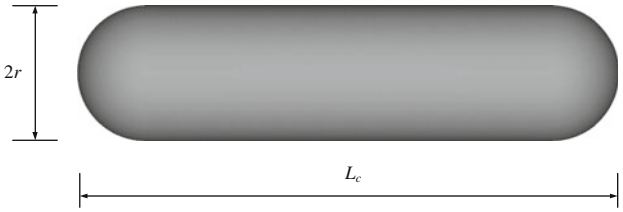


Fig. 13 Model of the fiber inclusion

Table 1 Parameters of the fiber inclusions

Case	L_c (m)	r (m)	κ_{fiber} (W/m K)	Number
A	40	5	1–10	1,000
B	40	2–8	5	1,000
C	20–70	5	5	1,000
D	52	6	1–10	1,000
E	100	6	1–10	500

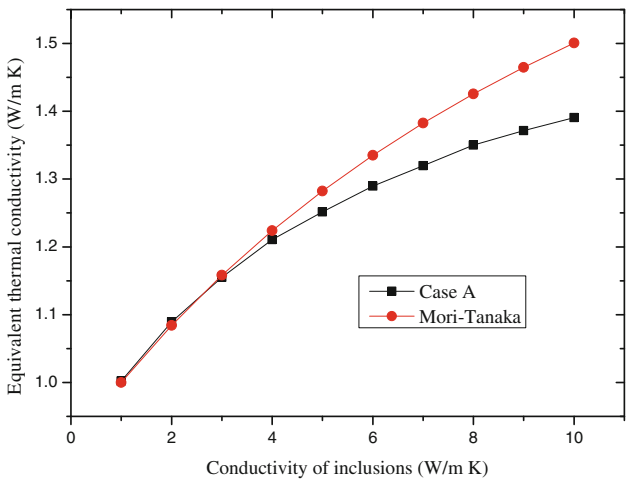


Fig. 14 Equivalent thermal conductivity with conductivity of fibers

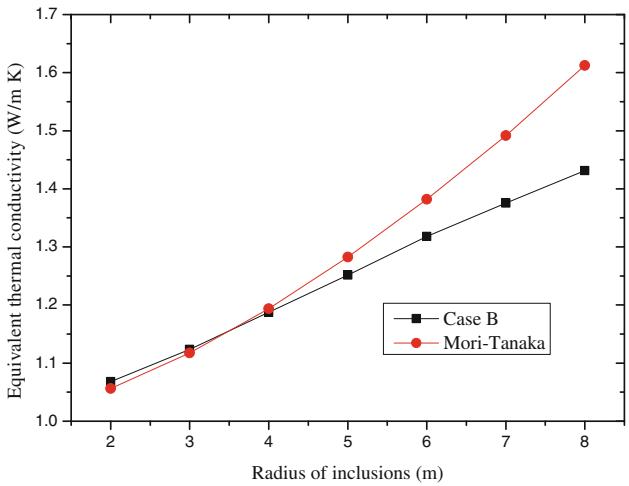


Fig. 15 Equivalent thermal conductivity with radius of fibers

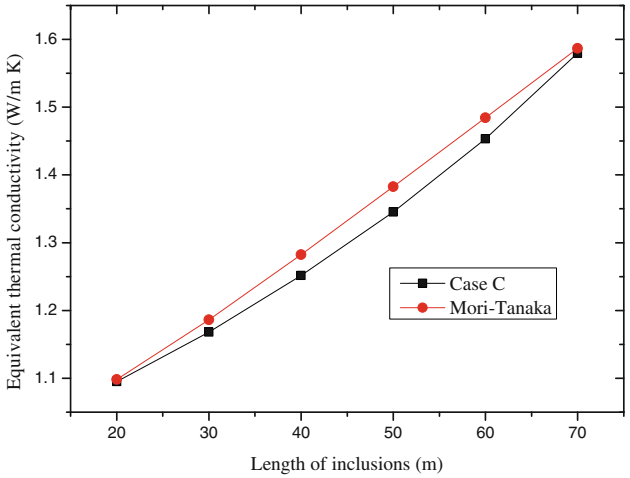


Fig. 16 Equivalent thermal conductivity with length of fibers

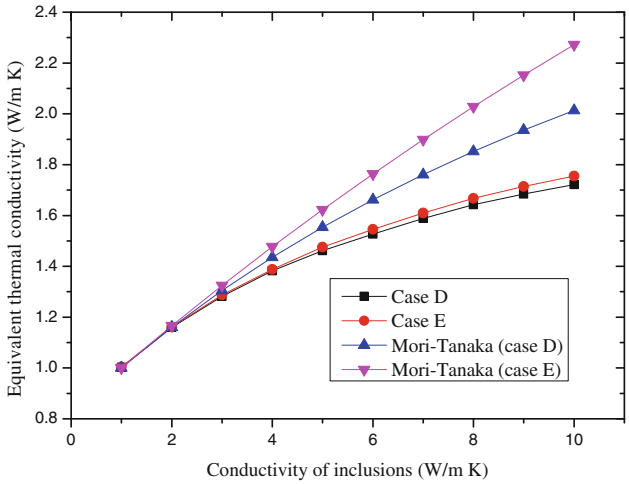


Fig. 17 Equivalent thermal conductivity with conductivity of fibers

If the fibers are randomly distributed

$$\bar{H}^j/\bar{H}^0 = \frac{1}{3}\sum_{i=1}^{3}\left(1+\frac{\kappa_j-\kappa_0}{\beta_i^{-1}\kappa_0}\right)^{-1} \tag{63}$$

If the fibers are aligned distributed along x_i axis

$$\bar{H}^j/\bar{H}^0 = \left(1+\frac{\kappa_j-\kappa_0}{\beta_i^{-1}\kappa_0}\right)^{-1} \tag{64}$$

where [38]

$$e = \sqrt{1-x_2^2/x_1^2} \tag{65}$$

$$\beta_1 = \frac{1-e^2}{2e^3}\left(\ln\frac{1+e}{1-e}-2e\right) \tag{66}$$

$$\beta_2 = \beta_3 = \frac{1}{2}(1-\beta_1) \tag{67}$$

The results in Figs. 14, 15 and 16 show that the equivalent thermal conductivity increases as the conductivity, the radius or the length of the fiber increases. Cases D and E are similar to case A. However, the fibers in case E are longer than case D and there are only 500 fibers in case E (see Fig. 12). Most important is that the volume fractions of the fibers in cases D and E are the same, and the value is 16.96 %. Figure 17 shows the results obtained from cases D and E and the results are also compared with Mori–Tanaka method. From Fig. 17 one can observe that the equivalent thermal conductivities of case E are a little higher than case D, and they become more obvious while the conductivities of the fibers increasing. We can also indicate that longer fibers are more efficient than shorter fibers when the volume fraction is the same.

In this example, one can observe that the results obtained by our method have difference with Mori–Tanaka method, one of the reasons is because we treat the fibers as spheroids while use Mori–Tanaka method. More research will be done to determine which method is more accurate or give a more accurate method. The CPU time cost by the new FM-HBNM is about one hour and more than 280 days will be used if use the conventional Hybrid BNM.

6 Conclusions

In this paper, the Hybrid BNM is applied for the large scale analysis of 3D composites. The thermal behavior of the composites with inclusions is studied by a new formulation, which can reduce the total DOFs in the final system equation than the conventional multi-domain solver.

The new version of FMM is coupled with the new formulation in order to apply the method to large scale analysis and the computational time can be reduced. The method presented in this paper retains the advantages of both the meshless method and the fast solver, which is especially applicable for large scale analysis of composites. The method proposed, which contains both the new formulation and new FM-HBNM, can be readily extended to other applications, such as large scale simulation of mechanical behavior in 3D composites.

As shown in the work of Yao et al. [39], the difference between the numerical and experimental results has been observed, it means that the numerical method has to be improved by considering more practical micro-structural factors, including the imperfect interface conditions and thermal contact between fibers. These researches will be investigated in the future.

Acknowledgments National Basic Research Program of China (973 Program: 2011CB013800) and Financial support for the Project from the Natural Science Foundation of Hubei Province, China (No. 2011CDB291).

References

1. Alexander A, Tzeng J (1997) Three dimensional effective properties of composite materials for finite element applications. J Compos Mater 31(5):466–485
2. Islam M, Pramila A (1999) Thermal conductivity of fiber reinforced composites by the fem. J Compos Mater 33(18):1699–1715
3. Nishimura N, Liu Y (2004) Thermal analysis of carbon-nanotube composites using a rigid-line inclusion model by the boundary integral equation method. Comput Mech 35(1):1–10
4. Chatterjee J, Henry D, Ma F, Banerjee P (2008) An efficient bem formulation for three-dimensional steady-state heat conduction analysis of composites. Int J Heat Mass Transf 51(5–6):1439–1452
5. Belytschko T, Lu YY, Gu L (1994) Element-free Galerkin methods. Int J Numer Methods Eng 37(2):229–256
6. Atluri SN, Zhu T (1998) A new meshless local Petrov–Galerkin (mlpg) approach in computational mechanics. Comput Mech 22(2):117–127
7. Mukherjee YX, Mukherjee S (1997) The boundary node method for potential problems. Int J Numer Methods Eng 40(5):797–815
8. Li XL, Zhu J (2009) A Galerkin boundary node method and its convergence analysis. J Comput Appl Math 230(1):314–328
9. Zhang JM, Qin XY, Han X, Li GY (2009) A boundary face method for potential problems in three dimensions. Int J Numer Methods Eng 80(3):320–337
10. Zhang JM, Yao ZH, Li H (2002) A hybrid boundary node method. Int J Numer Methods Eng 53:751–763
11. Zhang JM, Tanaka M, Matsumoto T (2004) Meshless analysis of potential problems in three dimensions with the hybrid boundary node method. Int J Numer Methods Eng 59:1147–1160
12. Miao Y, Wang YH (2005) Development of hybrid boundary node method in two-dimensional elasticity. Eng Anal Boundary Elem 29:703–712
13. Miao Y, Wang YH (2006) Meshless analysis for three-dimensional elasticity with singular hybrid boundary node method. Appl Math Mech 27(6):673–681
14. Miao Y, Wang Q, Liao BH, Zheng JJ (2009) A dual hybrid boundary node method for 2d elastodynamics problems. CMES Comput Model Eng Sci 53(1):1–22
15. Miao Y, Wang Y, Wang YH (2009) A meshless hybrid boundary-node method for Helmholtz problems. Eng Anal Boundary Elem 33(2):120–127
16. Wang Q, Zheng JJ, Miao Y, Lv JH (2011) The multi-domain hybrid boundary node method for 3d elasticity. Eng Anal Boundary Elem 35(6):803–810
17. Zhang JM, Tanaka M, Matsumoto T (2004) A simplified approach for heat conduction analysis of cnt-based nano-composites. Comput Methods Appl Mech Eng 193(52):5597–5609
18. Singh IV, Tanaka M, Endo M (2007) Thermal analysis of cnt-based nano-composites by element free Galerkin method. Comput Mech 39(6):719–728
19. Rokhlin V (1985) Rapid solution of integral equations of classical potential theory. J Comput Phys 60(2):187–207
20. Greengard L, Rokhlin V (1987) A fast algorithm for particle simulations. J Comput Phys 73(2):325–348
21. Zhang JM, Tanaka M, Endo M (2005) The hybrid boundary node method accelerated by fast multipole expansion technique for 3d potential problems. Int J Numer Methods Eng 63(5):660–680
22. Zhang JM, Tanaka M (2007) Systematic study of thermal properties of cnt composites by the fast multipole hybrid boundary node method. Eng Anal Boundary Elem 31(5):388–401
23. Zhang JM, Tanaka M (2008) Fast hdbnm for large-scale thermal analysis of cnt-reinforced composites. Comput Mech 41(6):777–787
24. Wang Q, Miao Y, Zheng JJ (2010) The hybrid boundary node method accelerated by fast multipole expansion technique for 3d elasticity. Comput Model Eng Sci 70(2):123–151
25. Wang Q, Miao Y, Zhu HP, Zhang C (2012) An O(N) fast multipole hybrid boundary node method for 3D elasticity. Comput Mater Continua 28(1):1–25
26. Wang Q, Miao Y, Zhu HP (2013) A fast multipole hybrid boundary node method for composite materials. Comput Mech 51:885–897
27. Kulkarni SS, Telukunta S, Mukherjee S (2003) Application of an accelerated boundary-based mesh-free method to two-dimensional problems in potential theory. Comput Mech 32(4):240–249
28. Liu YJ, Nishimura N, Yao ZH (2005) A fast multipole accelerated method of fundamental solutions for potential problems. Eng Anal Boundary Elem 29(11):1016–1024
29. Greengard L, Rokhlin V (1997) A new version of the fast multipole method for the laplace equation in three dimensions. Acta Numer 6(1):229–269
30. Yoshida K, Nishimura N, Kobayashi S (2001) Application of new fast multipole boundary integral equation method to crack problems in 3d. Eng Anal Boundary Elem 25(4):239–247
31. Nishimura N (2002) Fast multipole accelerated boundary integral equation methods. Appl Mech Rev 55:299
32. Shen L, Liu Y (2007) An adaptive fast multipole boundary element method for three-dimensional potential problems. Comput Mech 39(6):681–691
33. Bapat MS, Liu YJ (2010) A new adaptive algorithm for the fast multipole boundary element method. Comput Model Eng Sci 58(2):161–184
34. Saad Y, Schultz MH (1986) Gmres: a generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J Sci Stat Comput 7(3):856–869
35. Cheng H, Greengard L, Rokhlin V (1999) A fast adaptive multipole algorithm in three dimensions. J Comput Phys 155(2):468–498
36. Karris AN (1989) An examination of the mori-tanaka effective medium approximation for multiphase composites. J Appl Mech 56:83–88
37. Dunn M, Taya M, Hatta H, Takei T, Nakajima Y (1993) Thermal conductivity of hybrid short fiber composites. J Compos Mater 27(15):1493–1519
38. Landau LD, Lifšic EM, Sykes JB, Bell JS, Kearsley M, Pitaevskii LP (1960) Electrodynamics of continuous media, vol 364. Pergamon Press, Oxford
39. Yao ZH, Xu JD, Wang HT, Zheng XP (2009) Simulation of CNT composites using fast multipole BEM. J Mar Sci Technol 17:194–202