Applied Mathematical Modelling 39 (2015) 642–653

Contents lists available at ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

Generalization of the multi-scale finite element method to plane elasticity problems

L.X. Li *, Y.L. Chen [1], Z.C. Lu [1]

State Key Laboratory for Strength and Vibration of Mechanical Structures, Department of Engineering Mechanics, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, PR China

ARTICLE INFO

Article history:
Received 29 September 2013
Received in revised form 9 April 2014
Accepted 23 June 2014
Available online 2 July 2014

Keywords:
Plane elasticity
Multi-scale finite element method (MsFEM)
Shape functions
Accuracy and efficiency

ABSTRACT

In this paper, according to the governing differential equations of problem, the theory to construct the shape functions in the multi-scale finite element method is established for plane elasticity problems. An approach is then suggested to numerically solve the shape functions via the corresponding homogeneous governing equations on an element level. The linear, quadratic and cubic shape functions are finally obtained by prescribing the appropriate boundary conditions. Typical numerical experiments are conducted, including bending of a homogeneous beam, bending of a beam with voids, as well as bending of a beam with a random material distribution and with an oscillatory material property. The current work shows that the multi-scale finite element method has a prominent advantage in solution efficiency even for classic problems, and therefore can be implemented on a considerably coarse mesh for problems with complex microstructures, as well as for large scale problems to effectively save the solution cost.

1. Introduction

In recent decades, the conventional finite element method (CFEM) was greatly improved from many aspects. Typical examples include the numerical manifold method (NMM) proposed by Shi in 1991 [1], the generalized finite element method (GFEM) proposed by Babuska et al. in 1996 [2] and the extended finite element method (XFEM) proposed by Moes et al. in 1999 [3]. Although the shape functions are enriched in these methods for different purposes, the polynomial shape functions are still retained as the essential. In consequence, the shape functions are expressed either in terms of global area coordinates for triangular elements or in terms of local parent coordinates for quadrilateral elements (e.g. see [4,5]). Because the shape functions are a priori determined which are independent of problems to be solved, the solution accuracy and efficiency are often case dependent.

In virtue of this situation, for a problem with rough coefficients, Babuska et al. [6] proposed the idea that the basis functions adapted the specific problem, which was later developed by Hou et al. [7] to be the multi-scale finite element method (MsFEM). The MsFEM possesses numerical shape functions by solving a given problem, and therefore can, on one hand, capture the effect of microstructures on the macroscopic properties, and, on the other hand, obtain the response in a micro-scale via its shape functions. The convergence and the scale effect of the MsFEM have been verified from the mathematical

* Corresponding author. Tel.: +86 29 82668751; fax: +86 29 83237910.
E-mail address: luxianli@mail.xjtu.edu.cn (L.X. Li).
[1] Tel.: +86 29 82668751.

http://dx.doi.org/10.1016/j.apm.2014.06.012
0307-904X/

viewpoint [8,9]. The MsFEM was then successfully applied to many problems such as heat transfer in composite materials and porous media [7], the singularly perturbed convection–diffusion equation [10], the three-dimensional incompressible Navier–Stokes equations [11], and flow and transport equations [12]. Recently, the MsFEM was further developed to be the Generalized MsFEM (GMsFEM) to perform multiscale simulations for problems without scale separation over a complex input space [13].

In recent years, the MsFEM was developed to be the extended MsFEM (EMsFEM) by taking the Poisson's effect among different directions [14] to reduce the error in interpolating the vector field [15]. The EMsFEM was applied to more wide physical problems such as thermal conduction simulation in granular materials [16], elasto-plastic analysis [14,17], active response [18], 2D elastostatic analysis of heterogeneous materials [19], and 2D large displacement – small strain analysis of heterogeneous materials [20].

Motivated by Hou et al. [7] and Zhang et al. [14], the MsFEM is further generalized in this paper. In Section 2, through dissecting the plane elasticity problem, the theory is first established to construct the shape functions, and the boundary conditions are then discussed for the shape functions with appropriate orders. In Section 3, an effective approach is suggested to numerically solve the shape functions, and the partition of unity property is verified. In Section 4, the shape functions are applied to some examples to validate the accuracy and the efficiency. The concluding remarks are finally made in Section 5.

2. Basic theory

In this paper, plane elasticity problems are considered. In this case, the governing differential equations in stress form is

$$\sigma_{ij,j} + f_i = 0 \quad \text{in } \Omega, \tag{1}$$

where σ_{ij} are the components of stress tensor and $f_i(x, y)$ are the components of body force vector. The subscript, j denotes the partial differentiation with respect to coordinate x if $j = 1$ or y if $j = 2$. In addition, as a well-defined problem, over the whole boundary $\Gamma = \partial\Omega = \Gamma_u \cup \Gamma_t$, the essential boundary conditions and natural boundary conditions are prescribed respectively on boundary Γ_u and Γ_t, respectively.

After invoking the Hooke's law and the linear strain–displacement relations, Eq. (1) are eventually rewritten in terms of displacement field (u, v) as the following Navier equations [21]

$$\begin{cases} \nabla\cdot(\mu\nabla u) + \frac{\partial}{\partial x}[(\lambda+\mu)\theta] + f_x = 0, \\ \nabla\cdot(\mu\nabla v) + \frac{\partial}{\partial y}[(\lambda+\mu)\theta] + f_y = 0, \end{cases} \tag{2}$$

where λ and μ are the Lame constants, and may vary with position (x, y). Here, we assume that the material is isotropic but may be inhomogeneous and therefore the two constants vary with material points. $\theta = \partial u/\partial x + \partial v/\partial y$ is the displacement divergence (volume strain) in two dimensions.

For a linear problem, due to the superposition principle [21,22] and considering the basis feature of shape functions in the finite element method, the homogeneous form of Eq. (2) is fundamental, i.e.

$$\begin{cases} \nabla\cdot(\mu\nabla u) + \frac{\partial}{\partial x}[(\lambda+\mu)\theta] = 0, \\ \nabla\cdot(\mu\nabla v) + \frac{\partial}{\partial y}[(\lambda+\mu)\theta] = 0. \end{cases} \tag{3}$$

In the finite element method, over an element, the displacement is usually interpolated by

$$\begin{cases} u = \sum \phi_u^i u_i, \\ v = \sum \phi_v^i v_i, \end{cases} \tag{4}$$

where the subscript u or v is used to reflect the possible difference along the x and y directions for the shape functions.

Considering the versatility of the finite element method in approximating any possible variations, as a sufficient condition, on substituting Eq. (4) in Eq. (3), we stipulate that, on element level, the shape functions satisfy

$$\begin{cases} \nabla\cdot(\mu\nabla\phi_u^i) + \frac{\partial}{\partial x}\left[(\lambda+\mu)\frac{\partial\phi_u^i}{\partial x}\right] = 0, \\ \frac{\partial}{\partial y}\left[(\lambda+\mu)\frac{\partial\phi_u^i}{\partial x}\right] = 0 \end{cases} \tag{5a}$$

and

$$\begin{cases} \frac{\partial}{\partial x}\left[(\lambda+\mu)\frac{\partial\phi_v^i}{\partial y}\right] = 0, \\ \nabla\cdot(\mu\nabla\phi_v^i) + \frac{\partial}{\partial y}\left[(\lambda+\mu)\frac{\partial\phi_v^i}{\partial y}\right] = 0. \end{cases} \tag{5b}$$

It is easy to find that, in one sense, the shape functions ϕ_u^i and ϕ_v^i are not always identical in the present framework. In another sense, ϕ_u^i and ϕ_v^i are respectively governed by two different equations, i.e. Eq. (5a) or (5b), leading to be over-determinate. To get a closed-form solution, it is necessary to incorporate coupling terms in the interpolation of Eq. (4), i.e.

$$\begin{cases} u = \sum \phi_{uu}^i u_i + \sum \phi_{uv}^i v_i, \\ v = \sum \phi_{vu}^i u_i + \sum \phi_{vv}^i v_i, \end{cases} \tag{6}$$

where ϕ_{vu}^i is the coupling term to reflect the influence on the displacement v when the required boundary value u_i is prescribed (quite similar to the other coupling term ϕ_{uv}^i). Since a vector field problem governed by Eq. (1) is under study, the interpolations in Eq. (6) are more reasonable, as originally suggested by Zhang et al. [14].

Following the same procedure of deriving Eq. (5), we can eventually obtain governing equations for the shape functions over each element, i.e.

$$\begin{cases} \nabla \cdot (\mu \nabla \phi_{uu}^i) + \frac{\partial}{\partial x}\left[(\lambda+\mu)\left(\frac{\partial \phi_{uu}^i}{\partial x} + \frac{\partial \phi_{vu}^i}{\partial y}\right)\right] = 0, \\ \nabla \cdot (\mu \nabla \phi_{vu}^i) + \frac{\partial}{\partial y}\left[(\lambda+\mu)\left(\frac{\partial \phi_{uu}^i}{\partial x} + \frac{\partial \phi_{vu}^i}{\partial y}\right)\right] = 0, \end{cases} \tag{7a}$$

and

$$\begin{cases} \nabla \cdot (\mu \nabla \phi_{uv}^i) + \frac{\partial}{\partial x}\left[(\lambda+\mu)\left(\frac{\partial \phi_{uv}^i}{\partial x} + \frac{\partial \phi_{vv}^i}{\partial y}\right)\right] = 0, \\ \nabla \cdot (\mu \nabla \phi_{vv}^i) + \frac{\partial}{\partial y}\left[(\lambda+\mu)\left(\frac{\partial \phi_{uv}^i}{\partial x} + \frac{\partial \phi_{vv}^i}{\partial y}\right)\right] = 0. \end{cases} \tag{7b}$$

It is seen this time that Eq. (7) are closed and formally identical to the corresponding original Eq. (2), and can be regarded as a significant generalization of Eq. (2.4) in [7] for the MsFEM.

Apparently, it is hard to obtain the analytic solution to Eq. (7) except for very special cases (e.g. see [23]). To this end, a numerical solution via the CFEM on element level is preferred. Such a choice is in reality straightforward because the finite element code can be readily invoked when considering a strong resemblance between Eqs. (7) and (2), only using the specific boundary conditions instead.

Here, as an example, a quadrilateral element is employed to discuss the boundary conditions to be prescribed on Eq. (7) while bearing the properties of shape functions in mind [24].

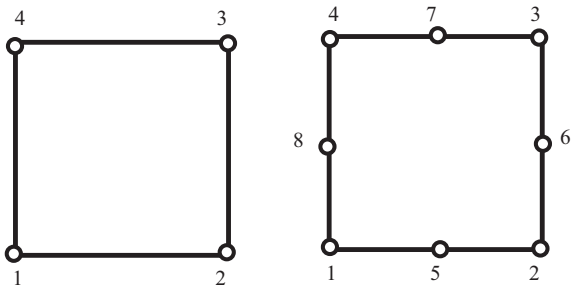
If two nodes are allocated at each element edge (see Fig. 1(a)), a linear compatibility will be built between the adjacent elements, and hence the boundary conditions (e.g. when solving shape functions ϕ_{uu}^1 and ϕ_{vu}^1 at node 1) for Eq. (7a) are

A1: ϕ_{uu}^1 varies linearly along 1–2 with $\phi_{uu}^1 = 1$ at node 1 and $\phi_{uu}^1 = 0$ at node 2,
A2: ϕ_{uu}^1 varies linearly along 1–4 with $\phi_{uu}^1 = 1$ at node 1 and $\phi_{uu}^1 = 0$ at node 4,
A3: ϕ_{uu}^1 vanishes along side 2–3,
A4: ϕ_{uu}^1 vanishes along side 3–4, and
A5: ϕ_{vu}^1 vanishes along all four sides.

Boundary conditions A1–A5 are called linear boundary conditions in [7].

If three nodes are allocated on each edge (see Fig. 1(b)), a quadratic compatibility will be built, and hence the boundary conditions (e.g. when solving shape functions ϕ_{uu}^1 and ϕ_{vu}^1 at node 1) for Eq. (7a) are

B1: ϕ_{uu}^1 varies quadratically along 1–2 with $\phi_{uu}^1 = 1$ at node 1 and $\phi_{uu}^1 = 0$, respectively at node 5 and node 2,
B2: ϕ_{uu}^1 varies quadratically along 1–4 with $\phi_{uu}^1 = 1$ at node 1 and $\phi_{uu}^1 = 0$, respectively at node 8 and node 4,



(a) 4-node element (b) 8-node element

Fig. 1. A quadrilateral element.

B3: ϕ^1_{uu} vanishes along side 2–6–3,
B4: ϕ^1_{uu} vanishes along side 3–7–4, and
B5: ϕ^1_{vu} vanishes for all four sides.

Similarly, we can get the corresponding boundary conditions for solving shape functions ϕ^i_{vv} and ϕ^i_{uv} from Eq. (7b).

Following the same procedure and by prescribing the corresponding boundary conditions, we can also get the third or higher order of shape functions.

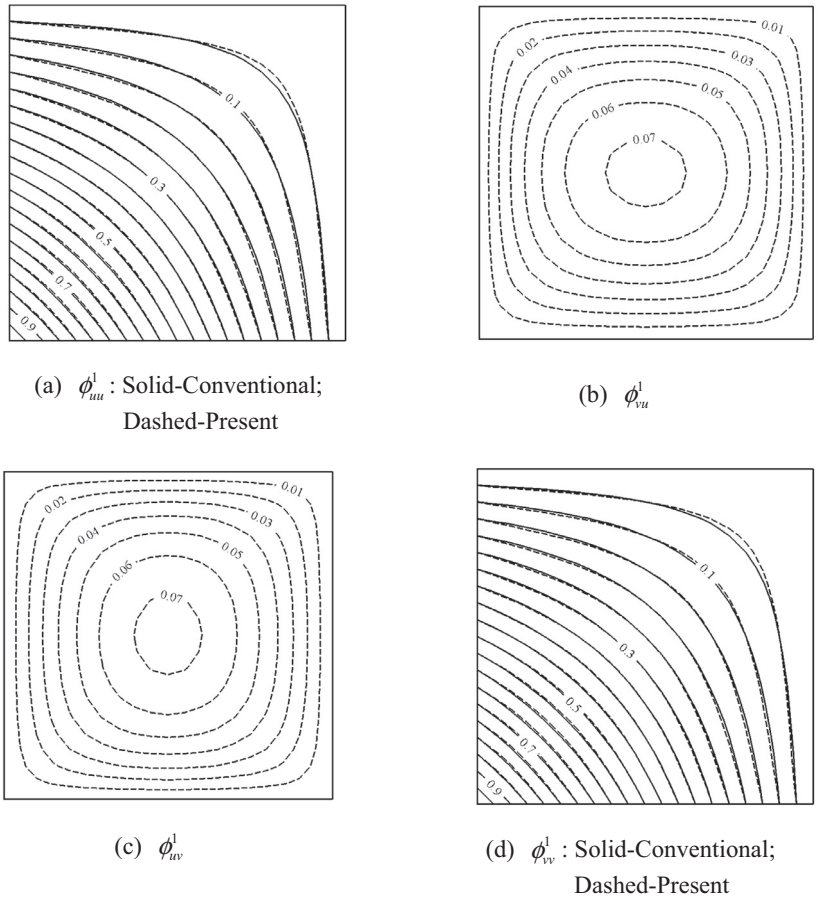
It should be noted that for an element with discontinuity along certain side, the boundary conditions can be piecewise linear along the side for the 8-node (see Fig. 1(b)) or 12-node, which will be reported in a separate paper.

3. The shape functions

3.1. Calculation of the shape functions

In this section, we shall calculate the shape functions for plane elasticity problems according to Eq. (7), and then make a comparison with the conventional one. Calculations will be carried out on element level by using the mapped 4-node quadrilateral CFEM with a finer mesh of $M \times M$. The element type may be of a 4-node (first order), 8-node (second order) or 12-node (third order) square. As is governed by Eq. (7), the coupling shape functions will reasonably appear for plane elasticity problems. It should be noted that the shape functions are suitable for all the element types with any complex geometry or materials inside the element. For the comparison reason, our attention is confined to a homogeneous continuum in this section.

First, we inspect the 4-node element. In Fig. 2 are shown the present shape functions ϕ^1_{uu} and ϕ^1_{vu} calculated from Eq. (7a), and ϕ^1_{vv} and ϕ^1_{uv} from Eq. (7b). It is found that the principal terms ϕ^1_{uu} and ϕ^1_{vv} are slightly deviated from the conventional bilinear shape functions. More interestingly, even for a homogeneous medium, the coupling terms ϕ^1_{vu} and ϕ^1_{uv} can achieve as big as 0.0735, rather than zero in the CFEM.



(a) ϕ^1_{uu} : Solid-Conventional; Dashed-Present

(b) ϕ^1_{vu}

(c) ϕ^1_{uv}

(d) ϕ^1_{vv} : Solid-Conventional; Dashed-Present

Fig. 2. The new shape functions for a square 4-node element at node 1.

For an 8-node element, the typical shape functions are shown in Figs. 3 and 4, respectively. It is seen that the present shape functions are quite different from the conventional ones even for the principal terms. Once again, we find the as big as 0.114 coupling terms even for a homogeneous medium. In addition, the shape functions ϕ^5_{uu} and ϕ^5_{vv} are no more identical (by comparing Fig. 4(a) and (d)) even for a square element.

The shape functions for a 12-node element can be calculated by following the same procedure. For the compact reason, typical ones are shown in Appendix, as compared with those in the CFEM if any.

3.2. Partition of unity property of the shape functions

By inspecting, it is found that the constant field $u = c$ and $v = 0$ is a solution to Eq. (3). So, if $u_i \equiv c$ and $v_i \equiv 0$, i.e. for a rigid body movement in x direction, the whole system will certainly have a respective movement, which implies that $u = c$ and $v = 0$. In light of the solution uniqueness of Eq. (3) and applying Eq. (6), we have

$$\begin{cases} 1 = \sum \phi^i_{uu}, \\ 0 = \sum \phi^i_{vu} \end{cases} \tag{8a}$$

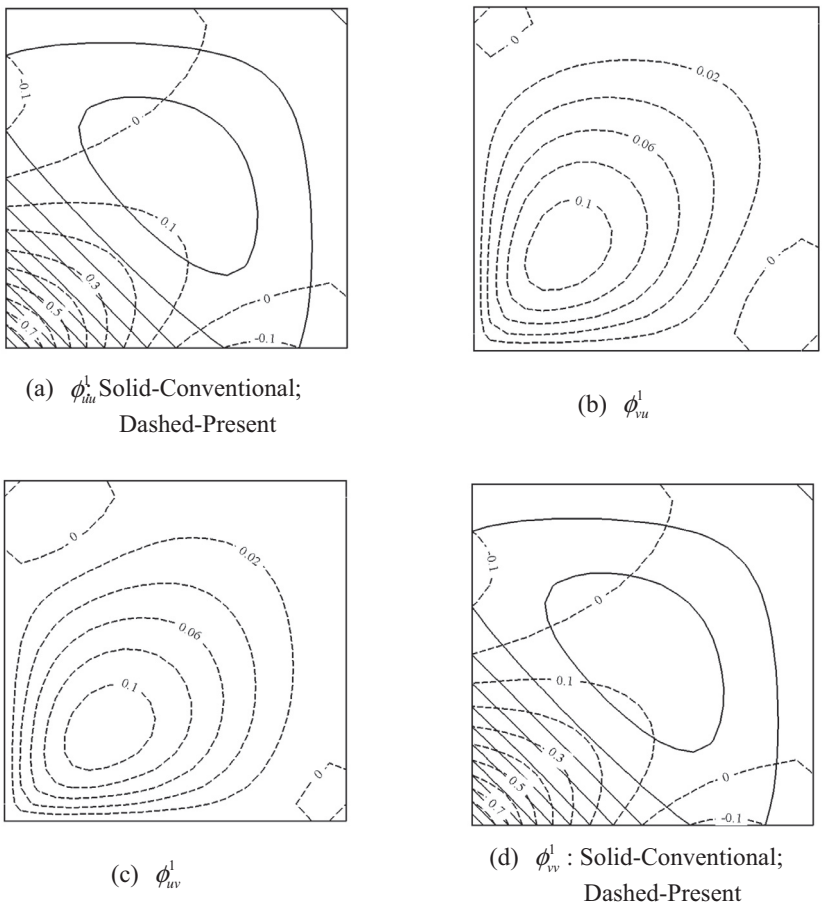
for 4-node, 8-node and 12-node elements.

In a similar way, by analyzing a rigid body movement in y direction, we also obtain

$$\begin{cases} 0 = \sum \phi^i_{uv}, \\ 1 = \sum \phi^i_{vv}. \end{cases} \tag{8b}$$

Eqs. (8a) and (8b) are termed the partition of unity property of the shape functions.

In fact, this property can also be numerically validated by substituting the present shape functions (e.g. Fig. 3 or 4) in Eq. (8a) or (8b).



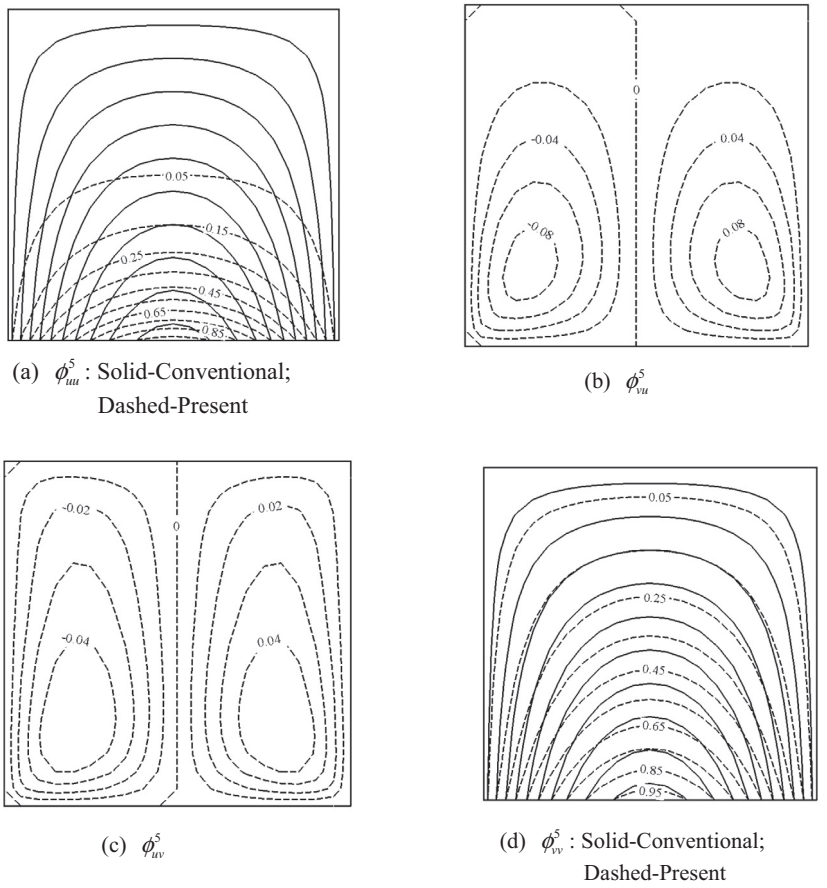
(a) ϕ^1_{uu} Solid-Conventional; Dashed-Present

(b) ϕ^1_{vu}

(c) ϕ^1_{uv}

(d) ϕ^1_{vv} : Solid-Conventional; Dashed-Present

Fig. 3. The new shape functions for a square 8-node element at node 1.



(a) ϕ_{uu}^5 : Solid-Conventional; Dashed-Present

(b) ϕ_{vu}^5

(c) ϕ_{uv}^5

(d) ϕ_{vv}^5 : Solid-Conventional; Dashed-Present

Fig. 4. The new shape functions for a square 8-node element at node 5.

4. Numerical examples

4.1. For accuracy check

Two examples are used to show the accuracy of the present method.

4.1.1. Bending of a homogeneous beam

First, we examine a cantilever beam subjected to a traction p_y at the right end. As shown in Fig. 5(a), the beam has dimensions of length L, height $h = L/5$ and width b, and is assumed to be fixed at the left end in the x direction and at the center of left end in the y direction.

In numerical calculation, a 10×2 mesh with rectangular 4-node (R4) elements (see Fig. 5(b)) is used. In the CFEM the conventional mapped shape functions are used while the shape functions are numerically calculated over each element in the present method. The deflection v along the neutral axis is analyzed. For the accuracy analysis, the L_2 norm error is calculated for the deflection v as

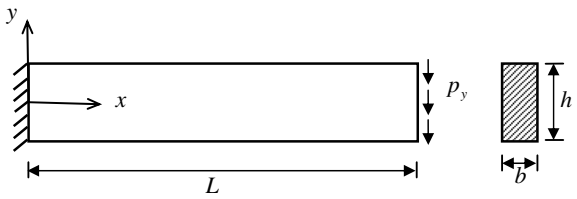
$$\mathrm{RE} = \frac{\sqrt{\int_{\Omega^h} \{v_{\mathrm{Ref}} - v_{\mathrm{Num}}\}^T \{v_{\mathrm{Ref}} - v_{\mathrm{Num}}\} \mathrm{d}\Omega}}{\sqrt{\int_{\Omega^h} \{v_{\mathrm{Ref}}\}^T \{v_{\mathrm{Ref}}\} \mathrm{d}\Omega}}, \tag{9}$$

where v_{Ref} and v_{Num} is the reference solution and the numerical result, respectively. For the case of $p_y = -0.75P(1 - y^2)$, the reference deflection to this problem is [25]

$$v = -\frac{P}{6EI}(3Lx^2 - x^3) - \frac{\nu P}{2EI}(L - x)y^2 - \frac{Ph^2}{8IG}x, \tag{10}$$

where E, ν, G and I are respectively the Young's modulus, Poisson's ratio, shear modulus and moment of inertia of the section.

When calculating the shape functions, a mesh of 6×6 is adopted for each element. The numerical results are plotted in Fig. 6 as compared with the reference. It is seen that the present method furnished with the shape functions alleviate, to



(a) A cantilever beam subjected to a traction

(b) A mesh with R4 elements

Fig. 5. A cantilever beam subjected to a traction and its numerical model.

some extent, the shear locking inherent in the bilinear shape functions. The L_2 norm errors are listed in Table 1. From this table we see that, due to the use of shape functions, the present method is more accurate than the CFEM.

4.1.2. Bending of a beam with voids

Fig. 7 shows a cantilever beam with uniformly distributed 10×2 small square voids. The beam has dimensions of length L, height $h = L/5$ and width b, and the voids have uniform dimensions of $L/30 \times L/30$. Assume that the beam be subjected to a uniform traction p_y at the right end, and clamped at the left end.

For the present method, a 10×2 mesh with rectangular 8-node elements is used with a square void inside. On account of the special micro-structure, the beam must however be divided at least into a 30×6 mesh with rectangular 8-node (R8) elements in the CFEM.

When calculating the shape functions, a mesh of 6×6 is adopted for each element. Since no analytic solution is available for this problem, the CFEM result from a finer mesh of 60×12 with R8 elements is used as the reference solution.

The L_2 norm errors expressed in Eq. (9) are summarized in Table 2 for the present method as compared with the CFEM. It is seen that the present method furnished with the new shape functions can achieve almost a bit better accuracy on a considerably coarser mesh (10×2) than the CFEM on a finer mesh (30×6).

4.2. For efficiency check

From Table 2, we also see that, for the CPU cost, the present method saves around six times (see the last column in Table 2) even when the likeness is not taken into account for each element. This indicates that the present method with the new shape functions can improve the solution efficiency along with alleviating the meshing task. To demonstrate this distinct feature, two typical examples are examined, which involve a larger scale of computations.

4.2.1. Bending of a beam with a random material distribution

We examine an inhomogeneous beam with same dimensions as in Fig. 7. However, the voids are replaced by the material with randomly distributed constant Young's modulus from 1.0×10^4 to 1.0×10^6 in each square element of a 160×32 mesh.

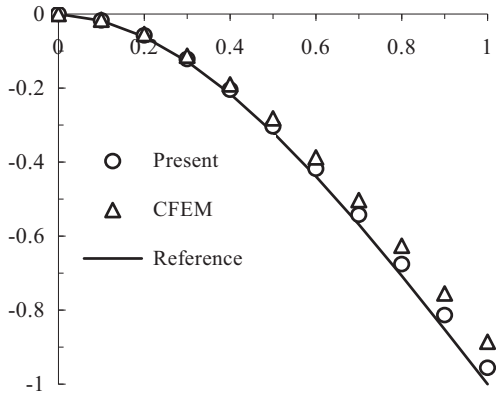


Fig. 6. The deflection of a uniform beam.

Table 1
Comparison of errors for bending of a homogeneous beam.

Method	Mesh	Error (%)
CFEM	10×2	11.89
Present	10×2	4.59

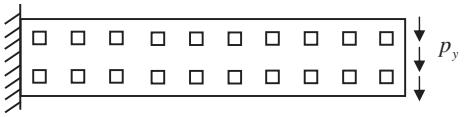


Fig. 7. A cantilever beam with 10×2 small voids.

Table 2
Comparison of accuracy and efficiency.

Method	Mesh	Error (%)	Cost (s)
R8-CFEM	30×6	1.55	2.156
R8-Present	10×2	1.18	0.328 (0.031)

Remark: The digit in the round bracket in Cost column (i.e. 0.031) is the CPU time for the case the new shape functions are calculated on any of the elements and then used for other elements due to the likeness of each element.

The Poisson's ration is 0.3. It is assumed that the left end be clamped and the center point of the right end be subjected to a load $P = 100$.

To solve this problem, comparably fine meshes are required. Here, the CFEM with R4 elements and the present method with R4 elements and R8 elements are respectively used. For the present method, calculation of the shape functions is carried out on a mesh of 8×8 over each element. For the comparison reason, the material distribution will be kept once it is randomly generated.

The CPU cost will be collected to compare the efficiency when the mesh is consecutively refined along each direction by parameter N. To this end, the reference cost (termed T1) for the three kinds of elements is obtained and listed in Table 3 for the reference mesh (termed xN1 $\times$ yN1). It should be noted that N1 in Table 3 is so determined that with these meshes the relative errors are around 1% each other. The ratio of the reference cost is around 1:0.054:0.124 for the R4-CFEM, R4-Present and R8-Present. With the reference cost, the logarithm increase in CPU cost (T) vs the logarithm refinement (N) is plotted in Fig. 8. It is seen that the CPU cost increases almost by a rate of 2^5 for the R4-CFEM (see the solid line in Fig. 8). However, for the R4-Present and R8-Present, the cost increases by a rate of 2^2 (see the two dashed lines in Fig. 8), only one-eighth as compared with the CFEM, with a very distinguished merit in computational efficiency even the difference in reference time is not taken into account yet.

Table 3
The reference CPU cost (T1) for the three kinds of elements.

Method	Mesh (xN1 $\times$ yN1)	T1(s)
R4-CFEM	160×32	29.406
R4-Present	20×4	1.594
R8-Present	20×4	3.656

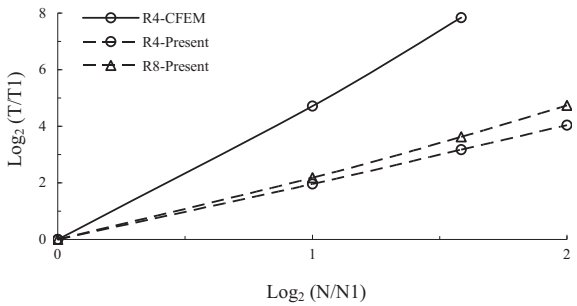


Fig. 8. Comparison of the solution efficiency for bending of a beam with a random material distribution.

Table 4
The reference CPU cost (T1) for the four kinds of elements.

Method	Mesh(xN1 × yN1)	T1(s)
R4-CFEM	100 × 20	4.516
R8-CFEM	50 × 10	6.422
R4-Present	10 × 2	0.906(0.078)
R8-Present	10 × 2	2.156(0.172)

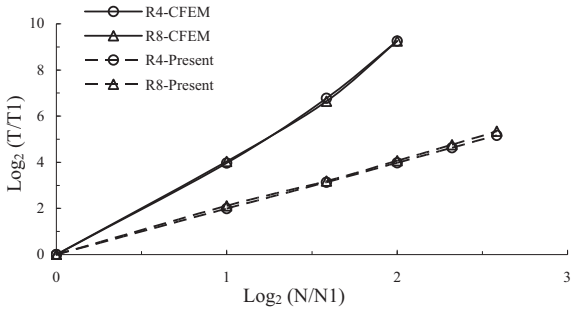


Fig. 9. Comparison of the solution efficiency for bending of a beam with an oscillatory material property.

4.2.2. Bending of a beam with an oscillatory material property

In this example, the Young's modulus oscillates continuously in the beam as

$$E(x,y)=\frac{1}{(a_0+p\cos(2\pi x/\varepsilon)(a_0+p\cos(2\pi y/\varepsilon)}, \tag{11}$$

where the constants are $a_0=0.0055$, $p=0.0045$ and $\varepsilon=0.25$, respectively. By Eq. (11), the Young's modulus is in the interval of $[1\times10^4, 1\times10^6]$. On the other hand, Equation (11) can also be understood as the overall property by using the Reuss estimate [26] along the x or y direction, respectively. Other specifications of the beam are same as that in Section 4.2.1.

To compare the solution efficiency, the CFEM and the present method with R4 and R8 elements are respectively used. In calculating the shape functions, a mesh of 10 × 10 is adopted for each element instead of a mesh of 8 × 8 in Section 4.2.1.

In contrast to Table 3, the reference CPU cost for the four kinds of elements is obtained and listed in Table 4. The ratio is 1:1.422:0.201:0.477 for R4-CFEM, R8-CFEM, R4-Present and R8-Present in turn.

With the reference cost, the logarithm increase in CPU cost (T) vs the logarithm refinement (N) is plotted in Fig. 9. It is seen that the CPU cost increases almost by a rate of 2^4 for the R4-CFEM and R8-CFEM. However, for the R4-Present and R8-Present, the cost increases by a rate of 2^2, only one-fourth as compared with the CFEM, with a very distinguished advantage in computational efficiency even the difference in reference time is not taken into account.

5. Conclusions and future work

A new idea to construct shape functions of the finite element method was proposed and the procedure to numerically calculate the corresponding shape functions was suggested together with the respective boundary conditions for various compatibilities. The numerical examples were given to show the accuracy and efficiency of the present method.

Since the shape functions can capture the geometric and physical features over an element level, the present method has following advantages:

First, furnished with the new shape functions, the present method possesses the higher accuracy even for classic problems (e.g. bending of a homogeneous beam).

Second, furnished with the new shape functions, for the same accuracy, the present method can be implemented in a considerably coarse mesh, and hence greatly reduce the computational cost.

Third, the current idea can be used to re-evaluate the apriori conventional shape functions, and higher orders of shape functions can be easily obtained in a straightforward manner.

It should be noted that in spite of capturing more micro-structural information in the shape functions, enough elements and more versatile (e.g. piecewise linear) compatibility along element edges are still required for the accuracy concern. In addition, numerical calculation of the new shape functions can be carried out on a general mesh, not necessary to adopt a regular or structured mesh in Section 4.

A similar approach has also been proposed by Hou et al. [7] from a mathematical viewpoint and by Zhang et al. [14] for coupling shape functions to reduce the error in interpolating the vector field [15]. However, in the current work, the origin of the new shape functions, the boundary conditions and the solution procedure are systematically dissected.

It is prospective that the present method can be applied to problems with more complicated microstructures or material properties, so as to conduct the multi-scale simulation by exactly interpolating the micro-scale field via the new shape functions. Application to three dimensional cases and elasto-plastic analysis is also straightforward. Furthermore, along this guideline, a key node finite element method (KN-FEM) has been developed for analysis of complex porous materials and the corresponding results will be reported.

Acknowledgments

This work was supported by the National Science Foundation of China (Grant Nos: 11272245, 11321062, 11025209).

Appendix A

The shape functions for a square 12-node (third order) element

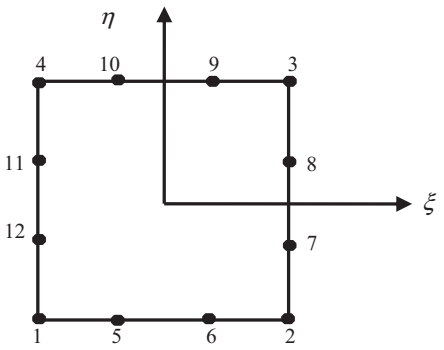


Fig. A1. The local numbering of a square 12-node element.

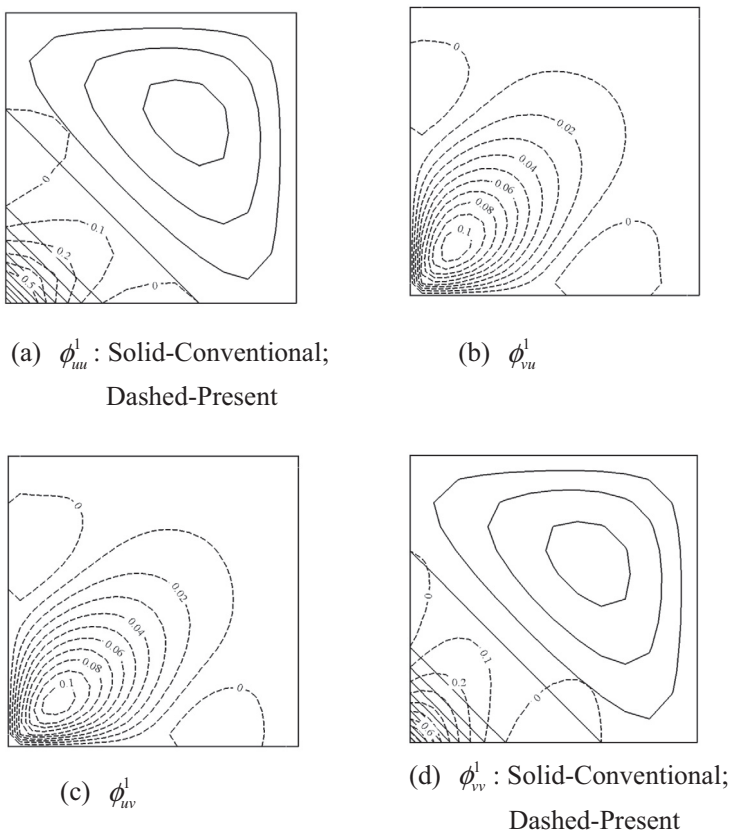
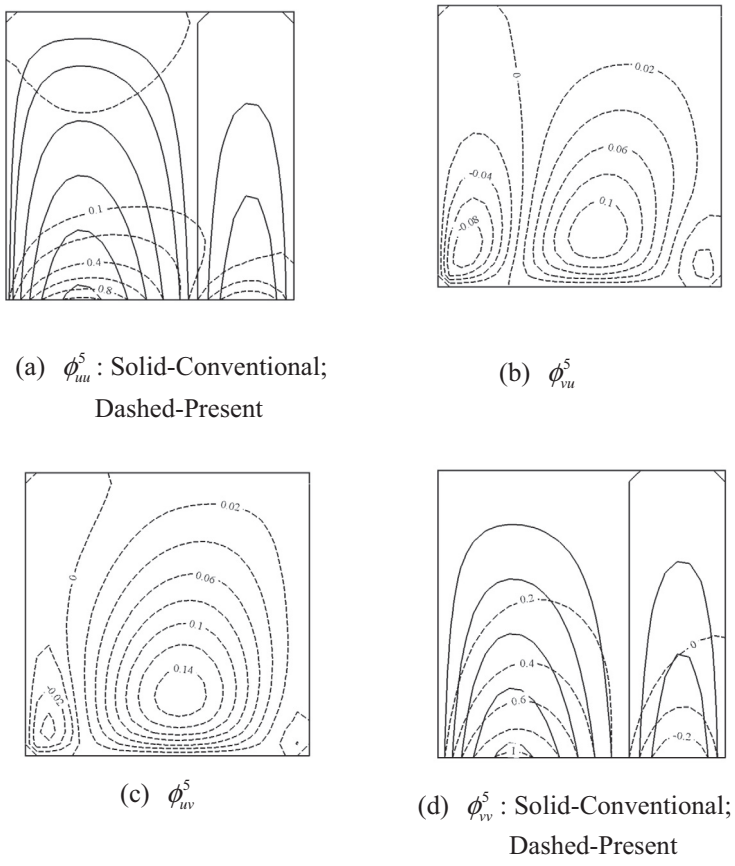


Fig. A2. The shape functions for a square 12-node element at node 1.



(a) ϕ_{uu}^5 : Solid-Conventional; Dashed-Present

(b) ϕ_{vu}^5

(c) ϕ_{uv}^5

(d) ϕ_{vv}^5 : Solid-Conventional; Dashed-Present

Fig. A3. The shape functions for a square 12-node element at node 5.

The 12 nodes in a 12-node square element are locally numbered as in Fig. A1. Following the guideline in Section 3.1, the shape functions for this element can also be calculated from Eq. (7) with the corresponding cubic compatibility conditions. As the representative, those at node 1 and 5 are shown in Figs. A2 and A3, respectively, as compared with the conventional ones if any.

References

[1] G.H. Shi, Manifold method of material analysis, in: Trans. Ninth Army Conf. on Applied Mathematics and Computing, Minneapolis, Minnesota, 1991, pp. 57–76.
[2] J.M. Melenk, I. Babuska, The partition of unity finite element method: basic theory and applications, Comput. Methods Appl. Mech. Eng. 139 (1996) 289–314.
[3] N. Moes, J. Dolbow, T. Belytschko, A finite element method for crack growth without remeshing, Int. J. Numer. Method Eng. 46 (1999) 131–150.
[4] L.X. Li, X.P. Han, S.Q. Xu, Study on the degeneration of quadrilateral element to triangular element, Commun. Numer. Methods Eng. 20 (2003) 671–679.
[5] L.X. Li, S. Kunimatsu, X.P. Han, S.Q. Xu, The analysis of interpolation precision of quadrilateral elements, Finite Elem. Anal. Des. 41 (1) (2004) 91–108.
[6] I. Babuska, G. Caloz, E. Osborn, Special finite element methods for a class of second order elliptic problems with rough coefficients, SIAM J. Numer. Anal. 31 (1994) 945–981.
[7] T.Y. Hou, X.H. Wu, A multiscale finite element method for elliptic problems in composite materials and porous media, J. Comput. Phys. 134 (1997) 169–189.
[8] T.Y. Hou, X.H. Wu, Z.Q. Cai, Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients, Math. Comput. 68 (227) (1999) 913–943.
[9] Y.R. Efendiev, T.Y. Hou, X.H. Wu, Convergence of a nonconforming multiscale finite element method, SIAM J. Numer. Anal. 37 (3) (2000) 888–910.
[10] P.J. Park, T.Y. Hou, Multiscale numerical methods for singularly perturbed convection–diffusion equations, Int. J. Comput. Methods 1 (1) (2004) 17–65.
[11] T.Y. Hou, D. Yang, H. Ran, Multiscale analysis and computation for the three-dimensional incompressible Navier–Stokes equations, Multiscale Model. Simul. 6 (4) (2008) 1317–1346.
[12] T.Y. Hou, D. Liang, Multiscale analysis for convection dominated transport equations, Discrete Contin. Dyn. Syst. 23 (1–2) (2009) 281–298.
[13] Y. Efendiev, J. Galvis, T.Y. Hou, Generalized multiscale finite element methods (GMsFEM), J. Comput. Phys. 251 (2013) 116–135.
[14] H.W. Zhang, J.K. Wu, Z.D. Fu, Extended multiscale finite element method for elasto-plastic analysis of 2D periodic lattice truss materials, Comput. Mech. 45 (2010) 623–635.
[15] H.W. Zhang, J.K. Wu, H. Liu, Z.D. Fu, Basic theory of extended multiscale finite element method, Comput. Aided Eng. 19 (2) (2010) 3–7 (in Chinese).
[16] H.W. Zhang, Q. Zhou, Y.G. Zheng, A multi-scale method for thermal conduction simulation in granular materials, Comput. Mater. Sci. 50 (10) (2011) 2750–2758.

[17] H.W. Zhang, J.K. Wu, J. Lv, A new multiscale computational method for elasto-plastic analysis of heterogeneous materials, Comput. Mech. 49 (2) (2012) 149–169.
[18] H.W. Zhang, J. Lv, A multiscale method for the numerical analysis of active response characterization of 3D nastic structures, Smart Mater. Struct. 21 (8) (2012) 085009.
[19] H. Liu, H.W. Zhang, A p-adaptive multi-node extended multiscale finite element method for 2D elastostatic analysis of heterogeneous materials, Comput. Mater. Sci. 73 (2013) 79–92.
[20] H. Liu, L. Zhang, D.S. Yang, H.W. Zhang, An efficient multiscale method for 2D large displacement-Small strain analysis of heterogeneous materials, Comput. Mater. Sci. 83 (2014) 443–456.
[21] H.S. Martin, Elasticity: Theory, Applications, and Numerics, Elsevier Butterworth-Heinemann, 200 Wheeler Road, Burlington, MA 01803, USA, 2009.
[22] S.P. Timoshenko, J.N. Goodier, Theory of Elasticity, McGraw-Hill Publishing Company, New York, 1970.
[23] H.W. Zhang, J.K. Wu, H. Liu, Y.G. Zheng, Z.D. Fu, Generalized plane and space rectangular elements, Chin. J. Comput. Mech. 27 (3) (2010) 391–396 (in Chinese).
[24] N. Sukumar, E.A. Malsch, Recent advances in the construction of polygonal finite element interpolants, Arch. Comput. Methods Eng. 13 (1) (2006) 129–163.
[25] X.C. Wang, The Finite Element Methods, Tsinghua University Press, Beijing, 2003 (in Chinese).
[26] L.X. Li, T.J. Wang, A unified approach to predict overall properties of composite materials, Mater. Charact. 54 (1) (2005) 49–62.