Development of the advanced software package based on S-FEM

R.P. Niu[1], *J.F. Zhang[1], †J.H. Yue[1], Y. Li[2]

[1] College of Mathematics, Taiyuan University of Technology, Taiyuan, Shanxi, 030024, China

[2] Department of Mathematics, City University of Hong Kong, Hong Kong SAR, China

*Presenting author: jfsufeng@gmail.com

† Corresponding author: woyuejunhong@163.com

Abstract: The paper reports a work to design and develop the advanced analysis software based on S-FEM. It supplies researchers and engineers a friendly graphical platform using various S-FEMs to solve 2D and 3D mechanics problems and heat transfer problems. The software package based on MFree 2D improves the original solver for 2D problems and develops a new solver for 3D problems. In order to enhance the meshing efficiency, we put forward to a parallelism algorithm based on OpenMP in pre-processing part. Through using the surface/line integrals method instead of the volume/area-weighted method used in MFree 2D, the system also works well for the quadrilateral elements. In the post-process part, all kinds of solutions are displayed by different figures according to users' requirement. Through numerical examples of 2D and 3D mechanics problems, we demonstrate our analysis software is efficient, friendly and accurate.

Key words: Analysis software; S-FEM; Mechanics problem; Smoothing domain

1 Introduction

Combining finite element method (FEM) and the meshless methods, G.R. Liu puts forward to a new novel computational method – smoothed finite element method (S-FEM) [1] which is based on G space theory [2][3]. S-FEM has several better properties compared with traditional FEM. For examples, it is known that FEM is a stiff model which leads to the low accuracy of strain solutions, while S-FEM has the softening effect to overcome the shortcoming [4]. Because of the softening effect, node-based smoothed finite method (NS-FEM) owns the upper bound property which can cooperate with the lower bound property of FEM to determine the range of the exact solution for practical mechanics problems [5]. In addition, S-FEM models are energy consistent when the assumed displacement is compatible along the boundaries of smoothing domains that ensures no energy loss in any of the violation of equilibrium. Besides, edge-based smoothed finite method (ES-FEM) often performs super convergence in both displacement norm and energy norm [6][7]. When different S-FEM models are combined in a unified scheme, they can solve some particular problems. For instance, ES/NS-FEM can effectively solve the volumetric locking problems [8].

More and more researchers are concentrating on studying S-FEM to solve the practical problems in different fields. In order to let researchers and engineers use S-FEM conveniently, G.R. Liu's team developed 2D S-FEM software, MFree 2D v2.0, in 2003 [9]. It supplies users a graphical user platform to solve the 2D mechanics problems using various S-FEM models, which includes ES-FEM, NS-FEM and FEM. In the implementation of MFree 2D, the triangular element is employed to interpolate the displacement field, which is because that there are automatic meshing programs for triangular mesh. Besides, the accuracy of the stress solution of ES-FEM using triangular mesh is as good as that of the FEM using the quadrilateral mesh. Just because of the linear interpolation used in MFree 2D, the real smoothing domains are not constructed actually. The simple area-weighted average formulation is applied to solve the smoothed strain-displacement matrix owing to that the compatible strain matrix in background triangular mesh is constant [10]. According to this scheme, we can solve 3D problems using volume-weighted average formulation for computing the smoothed strain matrix in background tetrahedral mesh. However, when we use quadrilateral elements or any other higher interpolation for displacement approximation in S-FEM, the simple method will not work. Therefore, we present a new method to solve the smoothed strain-displacement matrix.

In the paper, we design and develop the general and advanced software package based on S-FEM to solve 2D and 3D mechanics problems and heat transfer problems. Based on MFree 2D, we extend its original function to allow users to solve 3D problems. The software allows users to create parts in the user operating interface, or load the mesh file from ABAQUS and HyperMesh, which is very helpful for dealing with the problems with complicated geometry boundaries. In order to enhance the efficiency of meshing module, we design a parallel 3-noded triangular (Tr3) meshes generating code based on modified AFT algorithm for 2D problems. For 3D problems, because 3D meshing algorithms are often complicated, we use the robust and open source software –Tet-Gen for generating 4-noded tetrahedron (Te4) meshes [11]. According to the theory of S-FEM, the real smoothing domains are constructed and all the connectivities are built up and recorded in a database for the later computation [12][13], which is safer and more efficient compared with the data saved in the text storage. Then during the solving process, we put forward to the surface/line integrals method to compute the smoothed strain-displacement matrix strictly according to the general W^2 formulation of S-FEM, which is suitable for quadrilateral elements or any other higher interpolation. In the post-processing part, we basically keep the original functions of MFree 2D, and accomplish these functions for 3D S-FEM models.

2 The system frame

2.1 The pre-processing part

The pre-processing part is in charge of inputting the information about one project, which includes the problem geometry, the boundary constraints, the material properties, the mesh parameters and the chosen method. After we consider the task of the part and the relationship between different sections, the pre-processing part was designed to consist of the user interface module, the automatically meshing module and the database module.

(1) The users interface module
This module provides users an operating platform based on GUI (graphical user interface), shown in Figure 1. Users can build a new project, and then create parts using the graphical tools implemented in the tool bar or load an existent mesh obtained from other meshing software, such as ABAQUS. Next users can set the material properties and mesh parameters through dialog windows. There are four kinds of boundary constraints employed in the code, which are intensive loading, distributed loading, intensive displacement and distributed

displacement. Before executing the solver, the Tr3 background mesh must be completed by calling the automatic meshing module.

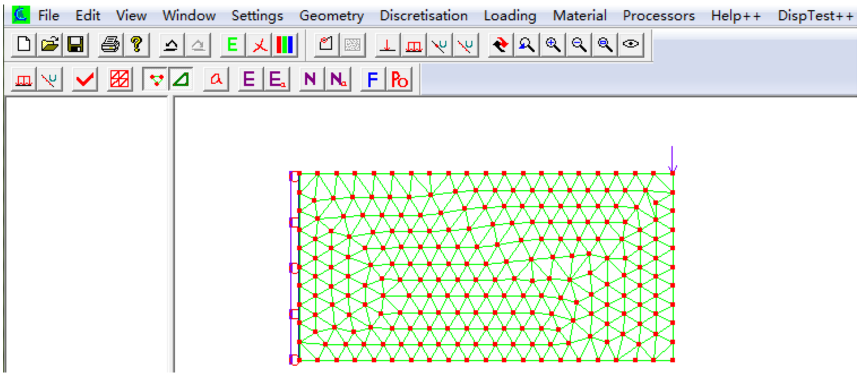


Figure 1 The user interface of the software package

(2) The automatic meshing module for 2D problem

It is well known that meshing is time-consuming, especially for 3D problems with complicated boundaries. We make use of the advancing front technique (AFT) to generate unstructured Tr3 mesh as the background mesh for S-FEM [14]. Through analyzing the algorithm, we find the structure of the code is suitable for parallel computing because of the weak data dependency. Hence, we develop a parallelism algorithm of modified AFT based on OpenMP method to fast generate mesh, which is proportional to the number of local CUP cores.

(3) The database module

The system can automatically create an independent access database for solving a problem. All the tables in the database will be arranged in a private space, which ensures the safety of data and improves the efficiency of the system. Through database management, the system can seek the data from tables quickly and directly which is more efficient than traversing the whole file in text storage.

2.2 The solver part

This part accomplishes different S-FEM solvers for 2D and 3D problems, which includes NS-FEM, ES-FEM, FS-FEM and even FEM treated as a special case of CS-FEM. The solvers strictly accord to the theory of S-FEM and follow the general procedure of S-FEM. We just list the steps which are not same as those conducted in MFree 2D. The rest steps can be in terms of the procedures introduce in [1].

(1) Creating smoothing domains

Multifarious smoothing domains are really constructed and all the segments of smoothing domain are recorded (a line-segment is an edge of a smoothing domain and a surface-segment is a face of a smoothing domain). At the same time, the related connectivities are saved in the database, which includes 12 relationships between node, edge, face and element. We compute the area of the surface-segment or the length of the line-segment, the volume or area of smoothing domains, the shape function values at the Gauss points and the unit-outward-normal vector of a line-segment or surface-segment for the later computation. The detail algorithms can be referenced in author’s other papers.

(2) Computing the smoothed strain matrix

In term of the formulation of S-FEM, we compute the smoothed strain matrix as the following equation

$$\bar{\mathbf{B}} = [\bar{\mathbf{B}}_1(\mathbf{x}), \bar{\mathbf{B}}_2(\mathbf{x}), ..., \bar{\mathbf{B}}_{N_n}(\mathbf{x})] \tag{1}$$

where its components are computed using

$$\bar{\mathbf{B}}_I(\mathbf{x}) = \begin{bmatrix} \frac{1}{A_k^s}\sum_{p=1}^{n_\Gamma^s} n_{x,p} N_I\left(\mathbf{x}_p^G\right) l_p & 0 \\ 0 & \frac{1}{A_k^s}\sum_{p=1}^{n_\Gamma^s} n_{y,p} N_I\left(\mathbf{x}_p^G\right) l_p \\ \frac{1}{A_k^s}\sum_{p=1}^{n_\Gamma^s} n_{y,p} N_I\left(\mathbf{x}_p^G\right) l_p & \frac{1}{A_k^s}\sum_{p=1}^{n_\Gamma^s} n_{x,p} N_I\left(\mathbf{x}_p^G\right) l_p \end{bmatrix} \quad for\ 2D$$

$$\bar{\mathbf{B}}_I(\mathbf{x}) = \begin{bmatrix} \frac{1}{V_k^s}\sum_{p=1}^{n_\Gamma^s} n_{x,p} N_I\left(\mathbf{x}_p^G\right) A_p^{surf} & 0 & 0 \\ 0 & \frac{1}{V_k^s}\sum_{p=1}^{n_\Gamma^s} n_{y,p} N_I\left(\mathbf{x}_p^G\right) A_p^{surf} & 0 \\ 0 & 0 & \frac{1}{V_k^s}\sum_{p=1}^{n_\Gamma^s} n_{z,p} N_I\left(\mathbf{x}_p^G\right) A_p^{surf} \\ \frac{1}{V_k^s}\sum_{p=1}^{n_\Gamma^s} n_{y,p} N_I\left(\mathbf{x}_p^G\right) A_p^{surf} & \frac{1}{V_k^s}\sum_{p=1}^{n_\Gamma^s} n_{x,p} N_I\left(\mathbf{x}_p^G\right) A_p^{surf} & 0 \\ 0 & \frac{1}{V_k^s}\sum_{p=1}^{n_\Gamma^s} n_{z,p} N_I\left(\mathbf{x}_p^G\right) A_p^{surf} & \frac{1}{V_k^s}\sum_{p=1}^{n_\Gamma^s} n_{y,p} N_I\left(\mathbf{x}_p^G\right) A_p^{surf} \\ \frac{1}{V_k^s}\sum_{p=1}^{n_\Gamma^s} n_{z,p} N_I\left(\mathbf{x}_p^G\right) A_p^{surf} & 0 & \frac{1}{V_k^s}\sum_{p=1}^{n_\Gamma^s} n_{x,p} N_I\left(\mathbf{x}_p^G\right) A_p^{surf} \end{bmatrix} \quad for\ 3D \tag{2}$$

where $n_{x,p}$ and $n_{y,p}$ (and $n_{z,p}$) are respectively the components of the unit-outward-normal vector of the *p*th line-segment for 2D (surface-segment for 3D). l_p is the length of the *p*th line-segment, and A_p^{surf} is the area of the *p*th surface-segment; n_Γ^s is the number of all line-segments for 2D (surface-segments for 3D) of the *k*th smoothing domain. A_k^s and V_k^s are, respectively, the area and the volume of the *k*th smoothing domain.

It is worth to notice that there is a local assembly to get the smoothed strain matrix **B**, which doesn't exist in FEM. That is because the size of the smoothed strain matrix **B** varies with the number of the support nodes of smoothing domain.

(3) Recovering the strain and stress

In order to recover the strain of an element in S-FEM, we use the 'raw' smoothed strains of all SDs sharing the element according to the below equations.

$$\begin{aligned} \tilde{\boldsymbol{\varepsilon}}_{ele_j} &= \frac{1}{\sum_{i=1}^{N} \bar{A}_i^s} \sum_{i=1}^{N} \bar{\boldsymbol{\varepsilon}}_{SD_i} \cdot \bar{A}_i^s \quad for\ 2D; \\ \tilde{\boldsymbol{\varepsilon}}_{ele_j} &= \frac{1}{\sum_{i=1}^{N} \bar{V}_i^s} \sum_{i=1}^{N} \bar{\boldsymbol{\varepsilon}}_{SD_i} \cdot \bar{V}_i^s \quad for\ 3D \end{aligned} \tag{3}$$

where $\tilde{\boldsymbol{\varepsilon}}_{ele_j}$ is the element strain of the *j*th element, *N* is a constant (*N*=3 for 2D ES-FEM; *N*=6 for 3D ES-FEM; *N*=3 for 2D NS-FEM; *N*=4 for 3D NS-FEM; *N*=4 for FS-FEM) and

$\bar{\boldsymbol{\varepsilon}}_{SD_i}$ is the smoothed strain of the ith SD which contains a part of the element in S-FEM.

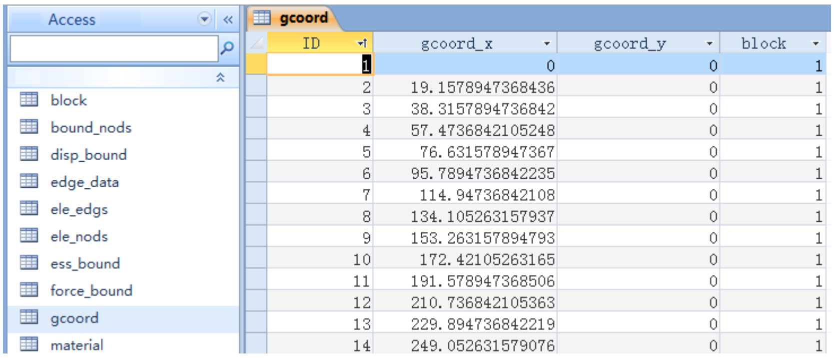
2.3 The post-processing part

This part is responsible for showing the displacement solutions, the strain solutions, the stress solutions and the energy solutions obtained from the solver, which are recorded in both a database and text files. We use three color modes – contour lines, color fields and smoother color fields to plot different figures of the solutions according to users' requirement. Besides, users can check the smoothing domains with the node ID or the smoothing domain ID. Through some graphical techniques, we make our post-processing interface friendly and figures plotted more smoothly and vividly.

3 numerical examples

3.1 2D cantilever beam

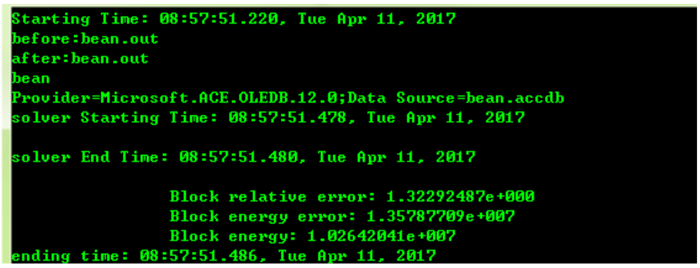
In this example, we consider a typical 2D mechanics problem – cantilever beam, which is fixed at its left end and subjected to a perpendicular force at the free end shown in Figure 1. The Tr3 background mesh is also found in Figure 1. At the same time, the mesh and project information are recorded in a database shown in Figure 2. There are many tables saving different information to ensure the independency of the data.



ID	gcoord_x	gcoord_y	block
1	0	0	1
2	19.1578947368436	0	1
3	38.3157894736842	0	1
4	57.4736842105248	0	1
5	76.631578947367	0	1
6	95.7894736842235	0	1
7	114.94736842108	0	1
8	134.105263157937	0	1
9	153.263157894793	0	1
10	172.42105263165	0	1
11	191.578947368506	0	1
12	210.736842105363	0	1
13	229.894736842219	0	1
14	249.052631579076	0	1

Figure 2 The database for the 2D cantilever beam project

Figure 3 is the messages in the background process which indicates the running status containing the starting time, the finishing time, and the strain energy and so on. Because this problem has the analytical solution, we can also compute the relative error of the displacement solution.



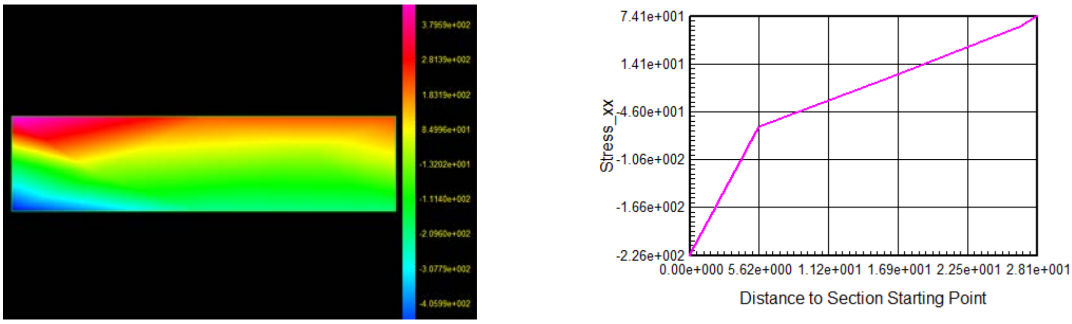
```
Starting Time: 08:57:51.220, Tue Apr 11, 2017
before:bean.out
after:bean.out
bean
Provider=Microsoft.ACE.OLEDB.12.0;Data Source=bean.accdb
solver Starting Time: 08:57:51.478, Tue Apr 11, 2017

solver End Time: 08:57:51.480, Tue Apr 11, 2017

                Block relative error: 1.32292487e+000
                Block energy error: 1.35787709e+007
                Block energy: 1.02642041e+007
ending time: 08:57:51.486, Tue Apr 11, 2017
```

Figure 3 The solving process using NS-FEM for the 2D cantilever beam problem

All the solutions can be displayed using the cloud atlas, contour and other figures. In Figure 4, we show the normal stress along x axis in the problem domain and on a special line user defined. It is easily found these figures show clearly and directly the solutions.

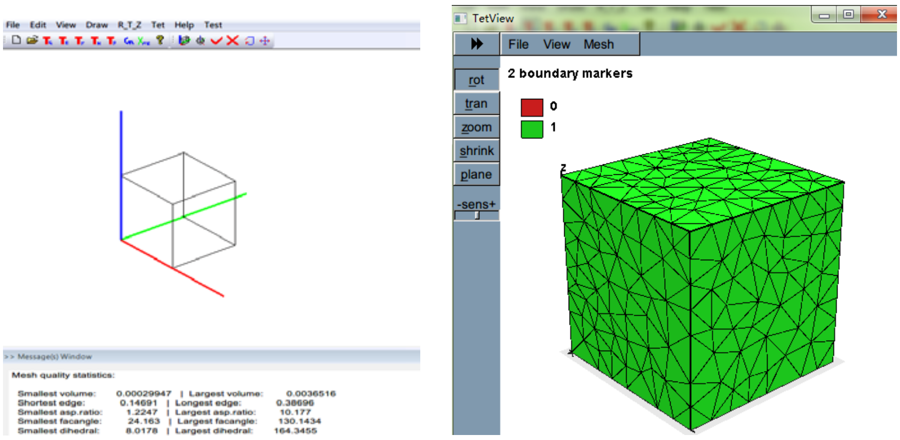


(a) (b)

Figure 4 The stress solutions. (a) The cloud atlas of the normal stress along x axis in the whole problem domain; (b)The values of the normal stress along x axis on a line designated by user.

3.2 3D cubic cantilever

Consider a 3D cubic cantilever to verify our program. The geometry settings are L=W=H=1. The cantilever is subjected to uniform pressure on its back surface and is fixed on its left surface. The parameters used in the experiment are E=1000N/m^2 and v=0.3.



(a) (b)

Figure 5 The pre-processing interface for 3D cubic cantilever. (a) The problem domain and meshing results; (b) the background Te4 mesh.

We use the meshing software Tet-Gen to generate the Te4 background mesh. As we know, 3D graphic processing is a very complicated field which is not our main tasks so that we also use the software TetView to display the meshing results. Figure 5 (a) is the pre-processing user interface in which the cubic cantilever is drawn and the meshing function is called. Besides, Figure 5 (b) is the Te4 background mesh displayed in TetView.

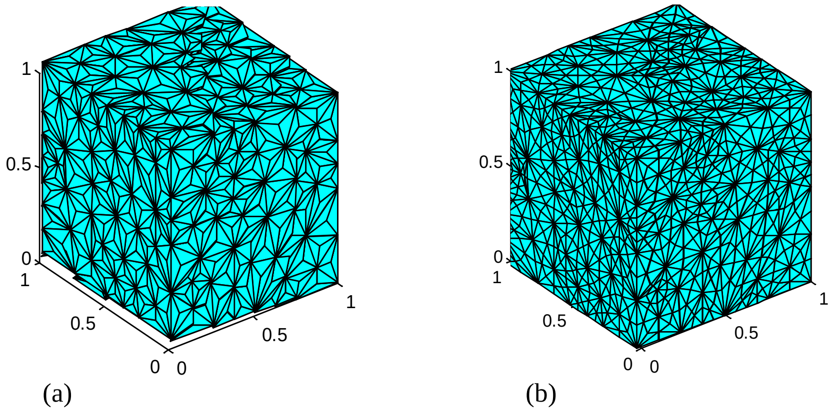


Figure 6 The smoothing domains for 3D cubic cantilever problem. (a) The edge-based smoothing domains; (b) The node-based smoothing domains

The different SDs can be given by the pre-processing interface, and the edge-based SDs and the node-based SDs for 3D cubic cantilever problem are shown in Figure 6. Figure 7 depicts the cloud atlas of the normal stress along *x* axis obtained using FEM, FS-FEM, ES-FEM and NS-FEM for 3D cubic cantilever problem.

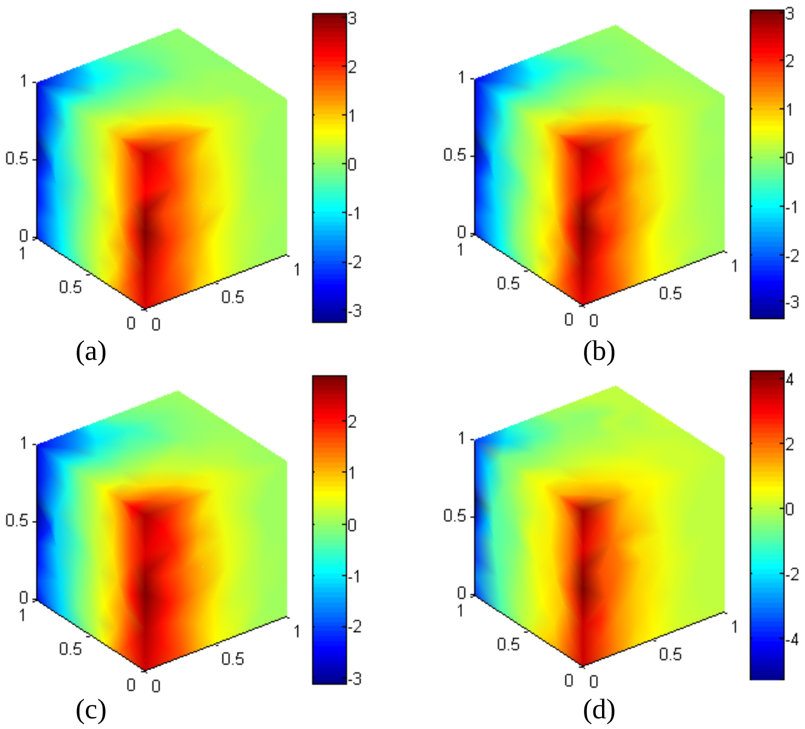


Figure 7 The cloud atlas of the normal stress solution along *x* axis using different numerical methods: (a) FEM; (b) FS-FEM; (c) ES-FEM; (d) NS-FEM for 3D cubic cantilever problem

Conclusions

In the paper, we briefly introduce our advanced software package based on S-FEM. The software provides users a graphical user interface to analyze the 2D and 3D mechanics problems or heat transfer problems using various S-FEM models. The user interface of our software package is very friendly and easy to operate. The solutions using any S-FEM model are accurate and efficient. Besides, the different solutions are displayed in many kinds of figures clearly and intuitively, and users can export solutions from the database for their further research.

References

[1] G.R. Liu, Nguyen Thoi Trung. Smoothed Finite Element Methods. CRC Press. 2010. 569-580

[2] G.R. Liu. A G space theory and a weakened weak (W2) form for a unified formulation of compatible and incompatible methods: Part I theory. International journal for numerical methods in engineering, 2010, 81: 1093-1126

[3] G.R. Liu. A G space theory and a weakened weak (W2) form for a unified formulation of compatible and incompatible methods: Part II applications to solid mechanics problems. International journal for numerical methods in engineering, 2010, 81: 1127-1156

[4] G.R. Liu, L. Chen, T. Nguyen-Thoi, et al. A novel singular node-based smoothed finite element method (NS-FEM) for upper bound solutions of fracture problems. International Journal for Numerical Methods in Engineering, 2010, 83(11); 1466-1497

[5] Cui Xiangyang, Li Guangyao, Zheng Gang, et al. NS-FEM/ES-FEM for contact problems in metal forming analysis. International Journal of Material Forming. 2010, 3: 887-890

[6] G.R. Liu, T. Nguyen-Thoi, K.Y. Lam. An edge-based smoothed finite element method (ES-FEM) for static, free and forced vibration analyses in solids. Journal of Sound and Vibration, 2009(320): 1100-1130

[7] T. Nguyen-Thoi, P.Phung-Van, T. Rabczuk, etc. An application of the ES-FEM in solid domain for dynamic analysis of 2D fluid-solid interaction problems. International Journal of Computational Methods, 2013, 10(01): 1-26

[8] G.R. Liu, T. Nguyen-Thoi, K.Y. Lam. A novel FEM by scaling the gradient of strains with factor α (αFEM). Computational Mechanics, 2009, 43: 369–391

[9] G. R. Liu. Meshfree Methods: Moving Beyond the Finite Element Method, 2nd edn. CRC Press, Boca Raton, USA, 2009.

[10] Wei Li, Yingbin Chai, Ming Lei, et al. Analysis of coupled structural-acoustic problems based on the smoothed finite element method (S-FEM). Engineering Analysis Boundary Elements, 2014, 42: 84-91

[11] A. TetGen Quality tetrahedral mesh generator and 3d delaunay triangulator. version 1.5, user's manual.

[12] Y. Li, M. Li, G.R. Liu. A modified triangulation algorithm tailored for the smoothed finite element method (S-FEM). International Journal of computational methods, 2014, 11(1):1-24

[13] Y. Li, J.H. Yue, R.P. Niu, et al. Automatic mesh generation for 3D smoothed finite element method (S-FEM) based on the weaken-weak formulation. Advances in Engineering Software, 2016, 99 (111-120)

[14] Peraire, J., Vahdati, M., Morgan, etc. Adaptive remeshing for compressible flow computations, J. Comput. Phys. 1987, 72: 449–446

Acknowledgments

The project is supported by the National Natural Science Foundation of China (Grant No. 11472184)

Biography

R.P. Niu has been working at College of Mathematics, Taiyuan University of Technology. She mainly researches on S-FEM and meshless methods. Now she is a PH.D candidate following Professor G.R. Liu.