

# BEM–EDM COUPLED ANALYSIS OF MULTI-SCALE PROBLEMS

YONG-TONG ZHENG, XIAO-WEI GAO & HAI-FENG PENG

State Key Laboratory of Structural Analysis for Industrial Equipment, Dalian University of Technology, China

## ABSTRACT

In this paper, the element differential method (EDM), a new numerical method proposed recently, is coupled with the boundary element method (BEM), a traditional numerical method, for solving general multi-scale heat conduction problems. The basic algebraic equations in BEM are formulated in terms of temperatures and heat fluxes, which are the same as those in EDM. So, when coupling these two methods, we do not need to transform the variables like the thermal loads into heat fluxes as done with the finite element method (FEM). The key task in the proposed coupled method is to use the temperature consistency condition and the flux equilibrium equation at interface nodes of the two methods to eliminate all BEM nodes except for those on the interfaces. The detailed elimination process is presented in this paper, which can result in the final system of equations without iteration. The coefficient matrix of the final coupled system is sparse, even though a small part is dense. The coupled method inherits the advantage of EDM in flexibility and computational efficiency and the advantage of BEM in the robustness of treating multi-scale problems. A numerical example is given to demonstrate the correctness of this coupled method.

*Keywords:* coupled analysis, boundary element method, element differential method, multi-scale problems.

## 1 INTRODUCTION

In mathematics, to solve the type of second-order partial differential equations (PDEs) like heat conduction problems, there are a number of numerical methods available. The frequently used methods are finite element method (FEM) [1]–[7], boundary element method (BEM) [8]–[15] and some of mesh free methods (MFM) [16]–[21]. FEM is the most widely used method due to its flexibility and applicability and most engineering problems can be solved by it. BEM only needs to discretize the boundary of the problem into elements, which not only can reduce the dimension, but also can easily simulate some special cases like crack and stress concentration [8], [12]. However, the final system of BEM has a dense coefficient matrix and the fundamental solutions are derived from the linear problems, which severely limits the application of BEM in large problems, nonlinear and nonhomogeneous problems [12], [15]. Different from mesh-dependent methods, MFM only needs a group of distributed nodes in the computational domain and barely needs elements. Therefore, MFM can easily simulate the problems with irregular geometries. However, MFM has the drawbacks of time consuming to form the global shape functions and difficulty to apply boundary conditions [16], [17].

For multi-scale problems, they have at least a large-scale part and a small-scale part in the computational domain. For example, considering a big block with side length of 50 mm and 0.5 mm coatings around the surface, its large-scale part is block and its small-scale parts are coatings.

When we analyse heat conduction using FEM, both the block and its coating should be discretized into elements. To avoid the ill-formed elements in the coating, the size of each element should be close to the thickness of the coating, which may result in too many elements in modelling the block mentioned above. That is, the thinner the coating, the greater the number of elements.



If we use BEM to solve this type of problem, the scale of different dimensions has less influence on the scale of elements, since we only need to discretize the boundary of the computational domain. However, if the block itself is too big, the total number of nodes will be so many so that the system with dense coefficient matrix to be solved will be very large. Both forming and solving the system are time consuming.

A better way to solve multi-scale problems is using BEM to analyse the coating, and FEM to analyse the block. Then the two systems are coupled together by temperature consistency and flux equilibrium on the interface. However, the basic algebraic equations in BEM are formulated in terms of temperatures and heat fluxes, while those in FEM are formulated in terms of temperatures and thermal loads. So, when coupling these two methods, we should transform the thermal loads into the heat fluxes, which need interpolation and integration [22].

Recently, a new numerical method called element differential method (EDM) is proposed by Gao et al. [23]–[25] and Cui et al. [26]. It is also a flexible method like FEM and can be used in most engineering problems. Crucially, EDM algebraic equations are formulated in terms of temperatures and heat fluxes, the same as with BEM. Therefore, it is easier to couple EDM and BEM together.

In this paper, the theory of EDM and BEM will be introduced briefly, while the coupling process of two methods is mainly illustrated. Finally, an example is given to verify the correctness of this coupled algorithm.

## 2 GOVERNING EQUATIONS FOR HEAT CONDUCTION PROBLEMS

The main research theme of this paper is multi-scale heat conduction and mechanical problems. Therefore, the governing equations of two problems are considered in the paper.

### 2.1 Steady heat conduction problems

For transient heat conduction problems, discretization in the time domain is exactly the same as with FEM and will not be considered here.

For steady heat conduction problems with a spatially varying thermal conductivity and heat source, the governing equation can be expressed as

$$\frac{\partial}{\partial x_i} \left( \lambda_{ij}(x) \cdot \frac{\partial T(x)}{\partial x_j} \right) + Q(x) = 0 \quad x \in \Omega, \quad (1)$$

and three types of boundary conditions are shown as follows:

$$T(x) = \bar{T} \quad x \in \Gamma_1, \quad (2)$$

$$-\lambda_{ij}(x) \cdot \frac{\partial T(x)}{\partial x_j} \cdot n_i(x) = \bar{q} \quad x \in \Gamma_2, \quad (3)$$

$$-\lambda_{ij}(x) \cdot \frac{\partial T(x)}{\partial x_j} \cdot n_i(x) = h(T(x) - T_\infty) \quad x \in \Gamma_3, \quad (4)$$

in which,  $\Omega$  is the computation domain of problem,  $\Gamma$  is the boundary of  $\Omega$ ,  $\Gamma = \Gamma_1 + \Gamma_2 + \Gamma_3$ ,  $x$  represents coordinate vector,  $n$  is the outward normal to the boundary  $\Gamma$ ,  $h$  is the heat transfer coefficient;  $\bar{q}$  and  $\bar{T}$  are the prescribed heat flux and temperature on the boundary,



respectively;  $T_\infty$  is the prescribed environment temperature;  $\lambda$  is the thermal conductivity tensor,  $T$  is the temperature, and  $Q$  is the heat-generation rate. The repeated subscripts  $i$  and  $j$  represent the summation over their range which is 2 for 2D and 3 for 3D problems.

### 3 ELEMENT DIFFERENTIAL METHOD

The main idea of EDM is to transform the physical variables and their first- and second-order partial derivatives in PDE into shape functions and their same order derivatives. Usually, we use the shape function of an isoparametric element [1], [2] because of its advantage in geometry representation and physical variable interpolation. In this section, it is shown briefly how to represent the physical variable derivatives by using shape function derivatives and assemble the final system.

#### 3.1 Computational process from physical variable derivatives to shape function derivatives

For the spatial coordinates and physical variables within the element, we can express them by shape function as:

$$x_i = N_\alpha x_i^\alpha, \quad (5)$$

$$T = N_\alpha T^\alpha, \quad (6)$$

where  $x_i^\alpha$  and  $T^\alpha$  denote the coordinates and temperatures at the  $\alpha$  node over an element, respectively;  $N_\alpha$  is the shape function at the node  $\alpha$ ; the repeated index  $\alpha$  represents the summation over all nodes in the element; the subscript  $i$  has values 1, 2 and 3 which represent  $x$ -,  $y$ - and  $z$ -directions for 3D problem.

The partial derivatives of physical variables shown in the governing equations can also be expressed by shape function:

$$\frac{\partial T}{\partial x_i} = \frac{\partial N_\alpha}{\partial x_i} T^\alpha, \quad (7)$$

in which  $\partial N_\alpha / \partial x_i$ , the shape function derivative with respect to global coordinates, can be transformed into them with respect to intrinsic coordinates:

$$\frac{\partial N_\alpha}{\partial x_i} = \frac{\partial N_\alpha}{\partial \xi_j} \cdot \frac{\partial \xi_j}{\partial x_i} = \frac{\partial N_\alpha}{\partial \xi_j} \cdot \left( \frac{\partial x_i}{\partial \xi_j} \right)^{-1}, \quad (8)$$

where  $\partial x_i / \partial \xi_j$  is the Jacobian matrix mapping from the global coordinate system  $x_i$  to the intrinsic coordinate system  $\xi_j$ , with  $(x_1, x_2) = (x, y)$  and  $(\xi_1, \xi_2) = (\xi, \eta)$  for 2D problems, and  $(x_1, x_2, x_3) = (x, y, z)$  and  $(\xi_1, \xi_2, \xi_3) = (\xi, \eta, \varsigma)$  for 3D problems. It should

be noticed that  $(\partial x_i / \partial \xi_j)^{-1}$  means inversed Jacobian matrix. According to eqn (5), all items in Jacobian matrix can be easily computed using the following expression:

$$[J]_{ij} = \frac{\partial x_i}{\partial \xi_j} = \frac{\partial N_\alpha}{\partial \xi_j} x_i^\alpha. \quad (9)$$



The equations above shows the computational process of the first-order derivatives. Second-order derivatives can be obtained by the same way as shown in the following equations:

$$\frac{\partial^2 T}{\partial x_i \partial x_j} = \frac{\partial^2 N_\alpha}{\partial x_i \partial x_j} T^\alpha, \quad (10)$$

$$\frac{\partial^2 N_\alpha}{\partial x_i \partial x_j} = \frac{\partial}{\partial x_j} \left( \frac{\partial N_\alpha}{\partial x_i} \right) = \frac{\partial}{\partial x_j} \left( \frac{\partial N_\alpha}{\partial \xi_k} \cdot [J]_{ik}^{-1} \right) = \left[ [J]_{ik}^{-1} \frac{\partial^2 N_\alpha}{\partial \xi_k \partial \xi_l} + \frac{\partial [J]_{ik}^{-1}}{\partial \xi_l} \frac{\partial N_\alpha}{\partial \xi_k} \right] [J]_{jl}^{-1}. \quad (11)$$

### 3.2 Assembling the system of equations based on governing equations and boundary conditions

When solving the heat conduction or mechanical problem by means of the EDM, the computational domain  $\Omega$  needs to be discretized into a series of elements that contain some nodes like FEM. The system of equations is established using a collocation scheme at each node. That is, one node matches only one equation and the nodes with different positions in the element have different equations. According to their positions, 3 types of nodes are defined as internal, interface and outer surface nodes. For example, in Fig. 1, there are four 9-node elements in the computational domain. The internal nodes refer to as the nodes located completely within elements like nodes number 7, 9, 17 and 19 in Fig. 1, and the interface nodes refer to as the nodes shared by a number of elements like nodes number 8, 12, 13, 14 and 18 in Fig. 1, whereas the outer surface nodes refer to as the nodes located on the outer boundary of the problem like the rest of the nodes not mentioned above in Fig. 1.

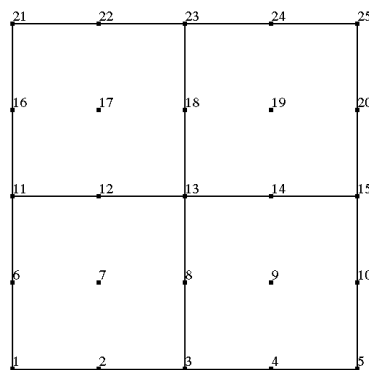


Figure 1: 9-node elements computational domain.

For the internal nodes, eqn (1) should be satisfied. Thus, by substituting eqns (7) and (10) into eqn (1), we can obtain the internal node equation for heat conduction problems:

$$\left[ \frac{\partial N_\beta(\xi)}{\partial x_i} k_{ij}^\beta \frac{\partial N_\alpha(\xi)}{\partial x_j} + k_{ij}(\xi) \frac{\partial^2 N_\alpha(\xi)}{\partial x_i \partial x_j} \right] T^\alpha + Q(\xi) = 0. \quad (12)$$

For the interface nodes, the traction equilibrium condition should be satisfied, i.e.,

$$\sum_{f=1}^M q^f(\xi_b) = 0, \quad (13)$$

where  $M$  is the number of element surface which the interface node is located on. Similarly, substituting eqns (7) and (10) into eqns (13), we can obtain the interface node equation for heat conduction problems (see eqn (14))

$$\sum_{f=1}^M -k_{ij}(\xi_b) \frac{\partial N_\alpha(\xi_b)}{\partial x_j} n_i^f(\xi_b) T^\alpha = 0. \quad (14)$$

For the outer surface nodes, the right-hand sides of eqn (13) are not 0, but  $\bar{q}$  or  $\bar{t}_i$ . Therefore, the equation for this kind of nodes become

$$\sum_{f=1}^M -k_{ij}(\xi_b) \frac{\partial N_\alpha(\xi_b)}{\partial x_j} n_i^f(\xi_b) T^\alpha = q(\xi_b). \quad (15)$$

Using the above equations, we can list a set of equations involving nodal temperatures and heat fluxes. Multiplying the specified values of temperatures and heat fluxes with their corresponding coefficients, a known vector  $\mathbf{b}$  can be formed, which is usually moved to the right-hand side of the system of equations. The other terms associated with unknowns are set on the left-hand side of the system of equations. Then, the final system equation can be written as:

$$[A]\{\mathbf{x}\} = \{\mathbf{b}\}, \quad (16)$$

where  $\mathbf{x}$  is a vector containing unknowns of each node;  $A$  is a sparse coefficient matrix.

By solving eqn (16) for vector  $\mathbf{x}$ , we can obtain all unknowns of the problem. That is all the solution process of heat conduction using EDM.

#### 4 BOUNDARY ELEMENT METHOD

The BEM is a widely known numerical method. So, the BEM theory is introduced briefly in this section.

For heat conduction problems, the boundary integral equation can be expressed as [8]:

$$c(P)T(P) + \int_{\Gamma} q^*(Q, P)T(Q)d\Gamma(Q) = \int_{\Gamma} T^*(Q, P)q(Q)d\Gamma(Q), \quad (17)$$

where  $c=1/2$  for smooth boundary points and  $c=1$  for interior points;  $P$  and  $Q$  represent the source points and field points, respectively;  $T$  and  $q$  are the temperature and heat flux defined on the boundary  $\Gamma$ ;  $T^*$  and  $q^*$  are the fundamental solutions for heat conduction problems, which are Green function and its derivative [27], [28].

For a BEM computational domain, we can just discretize its boundary into elements. Therefore, a 2D problem needs 1D line elements and a 3D problem needs 2D surface elements. Each node in the elements have their own boundary integral equation. Getting those equations together and moving the temperatures to left-hand side of the system and the heat fluxes to the right-hand side, we can obtain:

$$HT = Gq, \quad (18)$$

where  $H$  is the coefficient matrix of temperatures, and  $G$  is the coefficient matrix of heat fluxes. To solve the systems, the unknowns are moved to left-hand side of system and the



known values to the right-hand side as with EDM. Then the final system is obtained, whose format is like that of eqn (16).

### 5 COUPLING EDM AND BEM

When solving a multi-scale problem, the large-scale part of the computational domain would be better to be solved by EDM since it is flexible and efficient, while the small-scale part would be better to be solved by BEM because the latter method is robust in treating small scale problems. Then, we can couple these two methods using displacement/temperature consistency and traction/flux equilibrium to solve the multi-scale problems.

In order to couple these two methods, we should list the two systems obtained for their own parts. According to eqn (18), the system of BEM can be written as:

$$H_{ss}^1 T_s^1 + H_{sc}^1 T_c^1 = G_{ss}^1 q_s^1 + G_{sc}^1 q_c^1, \quad (19)$$

$$H_{cs}^1 T_s^1 + H_{cc}^1 T_c^1 = G_{cs}^1 q_s^1 + G_{cc}^1 q_c^1, \quad (20)$$

where subscript  $s$  represents self-boundary nodes; subscript  $c$  represents EDM and BEM common interface nodes; the superscript 1 represents that the coefficient matrix or vector belong to BEM. For example,  $q_c^1$  is the heat fluxes of EDM and BEM common interface nodes on the BEM computational domain.

The system of EDM can be written as:

$$\begin{bmatrix} A_{nn}^2 & H_{nc}^2 \\ A_{cn}^2 & H_{cc}^2 \end{bmatrix} \begin{bmatrix} x_n^2 \\ T_c^2 \end{bmatrix} = \begin{bmatrix} y_n^2 \\ y_c^2 + q_c^2 \end{bmatrix}, \quad (21)$$

where  $A$  represents the coefficient matrix of the unknowns  $x$  (unknown displacements and tractions);  $H$  represents the coefficient matrix of displacements;  $y$  represents the vector forming by multiplying the coefficients with the knowns and adding them; subscript  $c$  represents EDM and BEM common interface nodes; subscript  $n$  represents the other nodes in the EDM computational domain; the superscript 2 represents that the coefficient matrix or vector belong to EDM;  $q_c^2$  is the heat fluxes of EDM and BEM common interface nodes on the EDM computational domain.

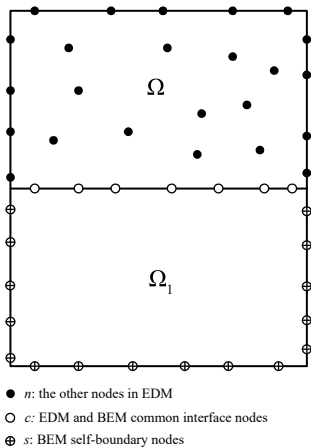


Figure 2: Three kind of nodes in computational domain.

Fig. 2 shows those kinds of nodes above directly. In Fig. 2,  $\Omega$  is EDM computational domain.  $\Omega_1$  is BEM computational domain.

For eqns (19) and (20), the unknowns of self-boundary nodes are moved to left-hand side of the system and the known values to the right-hand side and then the following equation can be obtained:

$$A_{ss}^1 x_s^1 + H_{sc}^1 T_c^1 = y_s^1 + G_{sc}^1 q_c^1, \quad (22)$$

$$A_{cs}^1 x_s^1 + H_{cc}^1 T_c^1 = y_c^1 + G_{cc}^1 q_c^1, \quad (23)$$

where  $A$  represents the coefficient matrix of unknowns  $x$ ;  $y$  represents the vector formed by multiplying the coefficients with the known values and adding them all.

From eqn (22), we can get:

$$x_s^1 = (A_{ss}^1)^{-1} (y_s^1 + G_{sc}^1 q_c^1 - H_{sc}^1 T_c^1). \quad (24)$$

Substituting the above equation into eqn (23), we obtain:

$$q_c^1 = (\overline{G_{cc}^1})^{-1} (\overline{H_{cc}^1} T_c^1 - \overline{y_c^1}), \quad (25)$$

where

$$\begin{cases} \overline{H_{cc}^1} = H_{cc}^1 - H_{cs}^1 (H_{ss}^1)^{-1} H_{sc}^1 \\ \overline{G_{cc}^1} = G_{cc}^1 - H_{cs}^1 (H_{ss}^1)^{-1} G_{sc}^1 \\ \overline{y_c^1} = y_c^1 - H_{cs}^1 (H_{ss}^1)^{-1} y_s^1. \end{cases} \quad (26)$$

According to displacement consistency condition and traction equilibrium equation, we can get:

$$q_c^1 + q_c^2 = 0, \quad (27)$$

$$T_c = T_c^1 = T_c^2. \quad (28)$$

Substituting eqns (25), (27) and (28) into the system of EDM (eqn (21)) yields

$$\begin{bmatrix} A_{nn}^2 & H_{nc}^2 \\ A_{cn}^2 & (\overline{G_{cc}^1})^{-1} \overline{H_{cc}^1} + H_{cc}^2 \end{bmatrix} \begin{bmatrix} x_n^2 \\ T_c \end{bmatrix} = \begin{bmatrix} y_s^2 \\ y_c^2 + (\overline{G_{cc}^1})^{-1} \overline{y_c^1} \end{bmatrix}. \quad (29)$$

Eqn (29) is the final system of the coupled method. The coefficient matrix  $(\overline{G_{cc}^1})^{-1} \overline{H_{cc}^1}$  is dense, while the other coefficient matrix is sparse. Usually, in the multi-scale problem, the number of EDM and BEM common interface nodes ( $c$ -nodes) is far less than that of the other nodes in EDM ( $n$ -nodes).

Therefore, the whole coefficient matrix is sparse even though a small part is dense.

After solving the system, we can use eqns (25) and (24) to calculate  $q_c^1$  and  $T_s^1$ . Thus, all the unknowns of computational domain can be obtained.

### 6 NUMERICAL EXAMPLES

We have already coded the coupled method using FORTRAN code. In this section, an example is computed by the codes to verify the correctness of the proposed method and code.

The problem to be considered is a 2D heat conduction plate with size of  $200 \times 100$ . We divide it into two same parts. The left part is computed by EDM, while the right one is computed by BEM. The EDM part is discretized into 100 9-node elements with 441 nodes, while BEM part is discretized into 40 3-node boundary elements with 80 boundary nodes as shown in Fig. 3. There are 21 common interface nodes in the whole computational domain. The left and right sides are insulated, while the top side is imposed with the temperature condition of 300 K and the bottom side is subjected to heat flux conditions of  $1 \text{ W/m}^2$ . The heat conductivity of the plate is  $1 \text{ W/(m}\cdot\text{K)}$ .

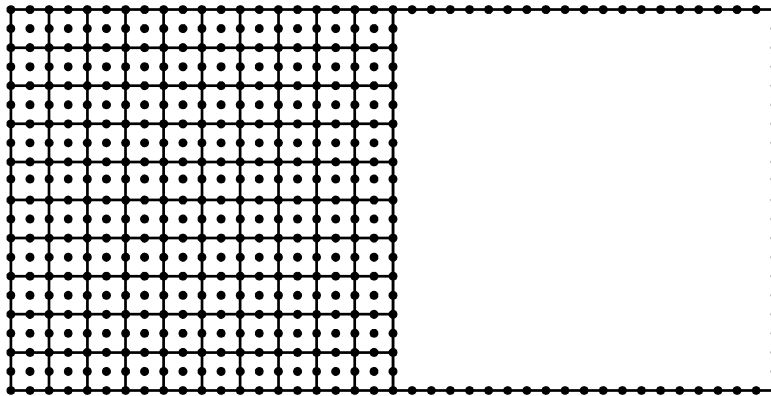


Figure 3: EDM and BEM Mesh over the plate.

After computing, we extract the temperatures of 21 common interface nodes and compare them to the exact solutions (see Fig. 4). From Fig. 4, it can be seen that the computed results are in good agreement with the exact solution. This means that both the coupled method and the FORTRAN code are correct.

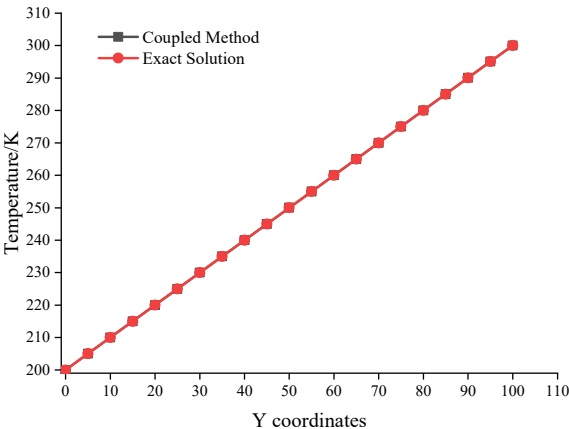


Figure 4: Computed temperatures of 21 common interface nodes and their exact solutions.



## 7 CONCLUSION

Through the above example, we can conclude that EDM-BEM coupled method can compute the heat conduction problem correctly. With the development of our research, the coupled method will be applied to the solution of more complicated engineering problems.

## ACKNOWLEDGEMENT

The support of this investigation by the National Natural Science Foundation of China (grant nos. 11672061, 11702054) is gratefully acknowledged.

## REFERENCES

- [1] Zienkiewicz, O.C. & Taylor, R.L., *The Finite Element Method*, 6th ed., Elsevier: London, 2005.
- [2] Hughes, T.J.R., *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*, Prentice-Hall: Englewood Cliffs, NJ, 1987.
- [3] Belytschko, T., *Nonlinear Finite Elements for Continua and Structures*, John Wiley, 2000.
- [4] Quek, G.R. & Liu, S., *The Finite Element Method: A Practical Course*, 2nd ed., Elsevier Science, 2013.
- [5] Gunzburger, M.D. & Bochev, P.B., *Least-Squares Finite Element Methods*, Springer: New York, 2009.
- [6] Lv, J., Sheng, G., Gao, X. & Zhang, H., Numerical Integration approach based on radial integration method for general 3D polyhedral finite elements. *International Journal of Computational Methods*, **12**(15500265), 2015.
- [7] Schroeder, J., Viebahn, N., Balzani, D. & Wriggers, P., A novel mixed finite element for finite anisotropic elasticity; the SKA-element simplified kinematics for anisotropy. *Computer Methods in Applied Mechanics and Engineering*, **310**, pp. 475–494, 2016.
- [8] Brebbia, C.A., *Boundary Element Techniques: Theory and Applications in Engineering*, Springer-Verlag: Berlin and New York, 1984.
- [9] Divo, E. & Kassab, A., *Boundary Element Method for Heat Conduction: With Applications in Non-Homogenous Media*, WIT Press, 2003.
- [10] Sladek, V., Sladek, J., Tanak, M. & Zhang, C., Transient heat conduction in anisotropic and functionally graded media by local integral equations. *Engineering Analysis with Boundary Elements*, **29**(11), pp. 1047–1065, 2005.
- [11] Xiao-Wei Gao, Hai-Feng Peng & Jian Liu, A boundary-domain integral equation method for solving convective heat transfer problems. *International Journal of Heat and Mass Transfer*, **63** pp. 183–190, 2013.
- [12] Gao, X.-W., Feng, W.-Z., Zheng, B.-J. & Yang, K., An interface integral equation method for solving general multi-medium mechanics problems. *International Journal for Numerical Methods in Engineering*, **107**(8), pp. 696–720, 2016.
- [13] Al-Jawary, M.A. & Wrobel, L.C., Radial integration boundary integral and integro-differential equation methods for two-dimensional heat conduction problems with variable coefficients. *Engineering Analysis with Boundary Elements*, **36**(5), pp. 685–695, 2012.
- [14] Liu, Y., *Fast Multipole Boundary Element Method: Theory and Applications in Engineering*, Cambridge University Press: Cambridge, 2009.
- [15] Gao, X.W., A meshless BEM for isotropic heat conduction problems with heat generation and spatially varying conductivity. *International Journal for Numerical Methods in Engineering*, **66**(9), pp. 1411–1431, 2006.



- [16] Gu, Y.T. & Liu, G.R., A meshfree weak-strong (MWS) form method for time dependent problems. *Computational Mechanics*, **35**(2), pp. 134–145, 2005.
- [17] Zhang, X.G., Liu, X.H., Song, K.Z. & Lu, M.W., Least-squares collocation meshless method. *International Journal for Numerical Methods in Engineering*, **51**(9), pp. 1089–1100, 2001.
- [18] Wang, D. & Wu, J., An efficient nesting sub-domain gradient smoothing integration algorithm with quadratic exactness for Galerkin meshfree methods. *Computer Methods in Applied Mechanics and Engineering*, **298**, pp. 485–519, 2016.
- [19] Wang, D. & Zhang, H., A consistently coupled isogeometric-meshfree method. *Computer Methods in Applied Mechanics and Engineering*, **268**, pp. 843–870, 2014.
- [20] Wen, P.H. & Aliabadi, M.H., An improved meshless collocation method for elastostatic and elastodynamic problems. *Communications in numerical methods in engineering*, **24**(8), pp. 635–651, 2008.
- [21] Zheng, B.-J., Gao, X.W., Yang K. & Zhang, C.Z., A novel meshless local Petrov–Galerkin method for dynamic coupled thermoelasticity analysis under thermal and mechanical shock loading. *Engineering Analysis with Boundary Elements*, **60**(SI), pp. 154–161, 2015.
- [22] Lin, C.-C., Lawton, E.C., Caliendo, J.A. & Anderson, L.R., An iterative finite element-boundary element algorithm. *Computers and Structures*, **59**(5), pp. 899–909, 1996.
- [23] Gao, X.-W., Liu, H.-Y., Xu, B.-B., Cui, M. & Lv, J., Element differential method with the simplest quadrilateral and hexahedron quadratic elements for solving heat conduction problems. *Numerical Heat Transfer Part B-Fundamentals*, **73**(4), pp. 206–224, 2018.
- [24] Gao, X.-W. et al., Element differential method for solving general heat conduction problems. *International Journal of Heat and Mass Transfer*, **115**(B), pp. 882–894, 2017.
- [25] Gao, X.-W. et al., Element differential method and its application in thermal-mechanical problems. *International Journal for Numerical Methods in Engineering*, **113**(1), pp. 82–108, 2018.
- [26] Cui, M., Xu, B.-B., Lv, J., Gao, X.-W. & Zhang, Y., Numerical solution of multi-dimensional transient nonlinear heat conduction problems with heat sources by an extended element differential method. *International Journal of Heat and Mass Transfer*, **126**(A), pp. 1111–1119, 2018.
- [27] Brebbia, C.A. & Dominguez, J., *Boundary elements: An introductory course*, Computational Mechanics Publications and McGraw-Hill, 1992.
- [28] Yu, B., Zhou, H.-L., Yan, J. & Meng, Z., A differential transformation boundary element method for solving transient heat conduction problems in functionally graded materials. *Numerical Heat Transfer, Part A: Applications*, **70**(3), pp. 293–309, 2016.

