Commun Nonlinear Sci Numer Simulat 16 (2011) 63–75

Contents lists available at ScienceDirect

Commun Nonlinear Sci Numer Simulat

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

The scaled boundary FEM for nonlinear problems

Zhiliang Lin, Shijun Liao *

State Key Laboratory of Ocean Engineering, School of Naval Architecture, Ocean and Civil Engineering, Shanghai JiaoTong University, Shanghai 200240, China

ARTICLE INFO

Article history:
Received 5 February 2010
Received in revised form 4 March 2010
Accepted 7 March 2010
Available online 15 March 2010

Keywords:
Scaled boundary finite-element
Homotopy analysis method
Nonlinear problem

ABSTRACT

The traditional scaled boundary finite-element method (SBFEM) is a rather efficient semi-analytical technique widely applied in engineering, which is however valid mostly for linear differential equations. In this paper, the traditional SBFEM is combined with the homotopy analysis method (HAM), an analytic technique for strongly nonlinear problems: a nonlinear equation is first transformed into a series of linear equations by means of the HAM, and then solved by the traditional SBFEM. In this way, the traditional SBFEM is extended to nonlinear differential equations. A nonlinear heat transfer problem is used as an example to show the validity and computational efficiency of this new SBFEM.

1. Introduction

The scaled boundary finite-element method (SBFEM), developed recently by Wolf and Song [1–3], is a novel semi-analytical method to solve partial differential equations. In the frame of the SBFEM, a scaled boundary coordinate system is introduced. The weighted residual approximation of finite elements is applied in the circumferential direction, and then the governing partial differential equations are transformed to ordinary differential equations in the radial direction. Those ordinary differential equations can be solved analytically in the radial direction. Like the boundary element method, only the boundary of the problem domain is discretized, but no fundamental solution is required. Thus, this semi-analytical method combines the advantages of the finite element and boundary element methods, and besides, it presents appealing features of its own.

The SBFEM has been used for the problems of elasto-statics and elasto-dynamics, especially for soil-structure interaction problems in unbounded domains [3,4]. Recently, the SBFEM has been extended to the fluid flow problems [5–8]. Meanwhile, researchers reported successful algorithms for the coupling of SBFEM with other numerical techniques [9–11] to analyze geomechanics problems. By coupling the finite element method and the SBFEM, Doherty and Deeks [9] captured the nonlinearity of problems in the near field. However, the SBFEM only accurately modeled the linear elastic far field response. It must be emphasized that, up to now, all problems solved by SBFEM are governed by linear partial differential equations.

The homotopy analysis method (HAM), which is an analytic approach to get series solutions of strongly nonlinear equations, was first proposed by Liao [12] in 1992. Thereafter, the HAM has been improved step by step [13–19]. Briefly speaking, by means of the HAM, one constructs a continuous mapping of an initial guess approximation to the exact solution of considered equations. An auxiliary linear operator is chosen to construct such kind of continuous mapping, and a convergence-control parameter is used to ensure the convergence of solution series. More importantly, this method enjoys great freedom in choosing initial guesses and auxiliary linear operators. By means of this kind of freedom, a complicated nonlinear problem can be transferred into an infinite number of simple, linear sub-problems, as shown by Liao and Tan [17]. Thus, if the SBFEM is successfully combined with the HAM, it could be extended to many engineering problems governed by nonlinear partial differential equations. This is the motivation of our present work.

* Corresponding author.
E-mail addresses: linzhiliang@sjtu.edu.cn (Z. Lin), sjliao@sjtu.edu.cn (S. Liao).

1007-5704/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.cnsns.2010.03.005

The aim of this paper is to combine the SBFEM with the HAM and then apply it to solve nonlinear problems. Shortly speaking, we first transfer an original nonlinear problem into an infinite number of linear sub-problems using the HAM, and then solve all those sub-problems by the SBFEM. To show its validity, 2D nonlinear Poisson-type problems are solved, and numerical results demonstrate the efficiency and accuracy of this approach.

2. Theoretical description

2.1. Governing equations

Consider a 2D Poisson equation with a varying conductivity $k(x,y)$, which can be expressed as

$$\nabla \cdot [k(x,y)\nabla u(x,y)] = f(x,y), \quad (x,y) \in \Omega, \tag{1}$$

under boundary conditions

$$u = g_1(x,y), \quad (x,y) \in \Gamma_1, \tag{2}$$

$$\frac{\partial u}{\partial n} = g_2(x,y), \quad (x,y) \in \Gamma_2, \tag{3}$$

where ∇ is the Hamilton operator, $u(x,y)$ is the dependent variable, the real functions $f(x,y)$, $g_1(x,y)$ and $g_2(x,y)$ are prescribed functions or given values, the x and y are spatial coordinates, n denotes the normal to the boundary, Ω the domain and $\Gamma = \Gamma_1 \cup \Gamma_2$ the boundary.

Obviously, Eq. (1) can be rewritten as

$$k(x,y)\nabla^2 u(x,y) + \nabla k(x,y) \cdot \nabla u(x,y) = f(x,y), \quad (x,y) \in \Omega. \tag{4}$$

Assuming that $k(x,y) = 1 + u(x,y)$, i.e., the conductivity is dependent on the unknown $u(x,y)$. Then Eq. (4) becomes

$$\nabla^2 u(x,y) + u(x,y)\nabla^2 u(x,y) + \nabla u(x,y) \cdot \nabla u(x,y) = f(x,y), \quad (x,y) \in \Omega. \tag{5}$$

Thus, the 2D nonlinear Poisson-type problems which we will solve are governed by Eq. (5) with the boundary conditions (2) and (3).

2.2. Homotopy transformation

According to Eq. (5), a nonlinear operator $\mathcal{N}$ is defined as follows:

$$\mathcal{N}[U] = \nabla^2 U + U\nabla^2 U + \nabla U \cdot \nabla U - f(x,y). \tag{6}$$

Then Eq. (5) can be written as

$$\mathcal{N}[u(x,y)] = 0, \quad (x,y) \in \Omega. \tag{7}$$

In the frame of the HAM, we first construct the zeroth-order deformation equation

$$(1-q)\mathcal{L}[U(x,y;q) - u_0(x,y)] = c_0 q \mathcal{N}[U(x,y;q)], \quad q \in [0,1], \ c_0 \neq 0, \ (x,y) \in \Omega \tag{8}$$

with the boundary conditions

$$U(x,y;q) = g_1(x,y), \quad (x,y) \in \Gamma_1, \tag{9}$$

$$\frac{\partial U(x,y;q)}{\partial n} = g_2(x,y), \quad (x,y) \in \Gamma_2, \tag{10}$$

where c_0 is the convergence-control parameter, $u_0(x,y)$ is an initial guess of $u(x,y)$, $\mathcal{L}$ is the auxiliary linear operator, and $q \in [0,1]$ is called homotopy-parameter, respectively. As q increases from 0 to 1, the solution $U(x,y;q)$ deforms continuously from the initial guess $u_0(x,y)$ to the solution $u(x,y)$ of Eqs. (5), (2) and (3).

The homotopy-series solution is given by

$$u(x,y) = u_0(x,y) + \sum_{m=1}^{+\infty} u_m(x,y), \quad (x,y) \in \Omega. \tag{11}$$

$u_m(x,y)$ $(m \geqslant 1)$ are governed by the so-called mth-order deformation equations

$$\mathcal{L}[u_m(x,y) - \chi_m u_{m-1}(x,y)] = c_0 \delta_m(x,y), \quad (x,y) \in \Omega, \tag{12}$$

with boundary conditions

$$u_m(x,y) = 0, \quad (x,y) \in \Gamma_1, \tag{13}$$

$$\frac{\partial u_m(x,y)}{\partial n} = 0, \quad (x,y) \in \Gamma_2, \tag{14}$$

where

$$\delta_m(x,y) = \frac{1}{(m-1)!}\frac{\partial^{m-1}\mathcal{N}[U(x,y;q)]}{\partial q^{m-1}}\bigg|_{q=0} \tag{15}$$

and

$$\chi_m = \begin{cases} 0, & m \leqslant 1, \\ 1, & \text{otherwise}. \end{cases} \tag{16}$$

According to Eqs. (6) and (15), we can obtain

$$\delta_m(x,y) = \nabla^2 u_{m-1}(x,y) + \sum_{j=0}^{m-1} u_j(x,y)\nabla^2 u_{m-1-j}(x,y) + \sum_{j=0}^{m-1} \nabla u_j(x,y)\cdot\nabla u_{m-1-j}(x,y) - (1-\chi_m) f(x,y). \tag{17}$$

Therefore, the original nonlinear problem governed by Eqs. (5), (2) and (3) has been transferred into an infinite number of linear sub-problems governed by Eqs. (12)–(14).

2.3. Implementation of SBFEM

It should be emphasized that, using the HAM, one has great freedom to choose the auxiliary linear operators and an initial guess under some basic rules [15,17]. Here, we select ∇^2 as the auxiliary linear operator. Then Eq. (12) can be rewritten as

$$\nabla^2 u_m(x,y) = R_m(x,y), \quad (x,y) \in \Omega, \tag{18}$$

where

$$R_m(x,y) = \chi_m \nabla^2 u_{m-1}(x,y) + c_0 \delta_m(x,y). \tag{19}$$

That is, the mth-order deformation Eq. (12) becomes a 2D linear Poisson equation and all the linear sub-problems become 2D linear Poisson-type problems which can be solved by the scaled boundary finite-element method.

To describe the implementation of the SBFEM, the 2D linear Poisson equation is written as

$$\nabla^2 u(x,y) = R(x,y) \quad \text{in } \Omega \tag{20}$$

with boundary conditions

$$u = \bar{u} \quad \text{on } \Gamma_u, \tag{21}$$

$$\frac{\partial u}{\partial n} = \bar{u}_n \quad \text{on } \Gamma_v, \tag{22}$$

where $R(x,y)$ is the so-called non-homogeneous term and the overbar denotes a prescribed value. The weak form of the weighted-residual statement is formulated as

$$\int_\Omega \nabla^T w \nabla u \,\mathrm{d}\Omega + \int_\Omega wR \,\mathrm{d}\Omega - \int_{\Gamma_v} w\bar{u}_n \,\mathrm{d}\Gamma = 0, \tag{23}$$

where w is the weighting function, and it is the starting point of the derivation of the SBFEM.

In the frame of the SBFEM, a scaled boundary coordinate system is introduced at first. As shown in Fig. 1, the circumferential coordinate s is anticlockwise along the boundary and the dimensionless radial coordinate ξ is a scaling factor, defined as 0 at the scaling centre O and 1 at the boundary S. The two straight sections $s = s_0$ and $s = s_1$ are termed side-faces. If the boundary is closed, the side-faces will coincide. Thus, the whole solution domain Ω is in the range of $\xi_0 \leqslant \xi \leqslant \xi_1$ and $s_0 \leqslant s \leqslant s_1$. Bounded domains containing the scaling centre are modeled by taking $\xi_0 = 0$ and $\xi_1 = 1$; whereas, unbounded domains are modeled by taking $\xi_0 = 1$ and $\xi_1 = \infty$.

The mapping between the Cartesian coordinate system and this scaled boundary coordinate system is expressed by the scaling equations

$$x = x_0 + \xi \cdot x_s(s), \quad y = y_0 + \xi \cdot y_s(s), \tag{24}$$

where (x_0, y_0) is the scaling center and $x_s(s)$ and $y_s(s)$ are C^0 continuous piecewise smooth functions describing the boundary S in Cartesian coordinates relative to the scaling center. Using conventional techniques the operator ∇ can be mapped to the scaled boundary coordinate system as [3]

$$\nabla = \mathbf{b}_1(s)\frac{\partial}{\partial \xi} + \frac{1}{\xi}\mathbf{b}_2(s)\frac{\partial}{\partial s}, \tag{25}$$

where $\mathbf{b}_1(s)$ and $\mathbf{b}_2(s)$ are dependent only on the definition of the boundary S:

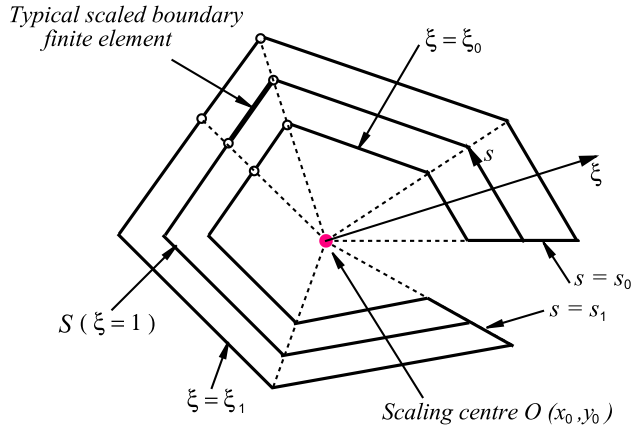

Fig. 1. The scaled boundary coordinate definition.

$$\mathbf{b}_1(s)=\frac{1}{|J|}\begin{Bmatrix} y_s(s)_{,s} \\ -x_s(s)_{,s}\end{Bmatrix},\quad \mathbf{b}_2(s)=\frac{1}{|J|}\begin{Bmatrix} -y_s(s) \\ x_s(s)\end{Bmatrix}, \tag{26}$$

and $|J|$ is the Jacobian at the boundary

$$|J|=x_s(s)y_s(s)_{,s}-y_s(s)x_s(s)_{,s}. \tag{27}$$

In the SBFEM, the unknown variables u at any point (ξ,s) in the domain are sought in the form

$$u(\xi,s)=\mathbf{N}(s)\mathbf{a}(\xi), \tag{28}$$

where $\mathbf{N}(s)$ is the shape function, the vector $\mathbf{a}(\xi)$ is analogous to the nodal values in the standard finite-element method. The weighting function w can be chosen as the same shape function as (28) by Galerkin approach

$$w(\xi,s)=\mathbf{N}(s)\mathbf{w}(\xi)=\mathbf{w}(\xi)^T\mathbf{N}(s)^T. \tag{29}$$

Substituting Eqs. (25), (28) and (29) into Eq. (23) and integrating the domain integrals containing $\mathbf{w}(\xi)_{,\xi}$ by parts with respect to ξ using Green's theorem, and noting that $\mathrm{d}\Omega=|J|\xi\,\mathrm{d}\xi\,\mathrm{d}s$, yields

$$\mathbf{q}(\xi_1)=\int_S \mathbf{N}(s)^T\bar{u}_n(\xi_1,s)\xi_1\,\mathrm{d}s, \tag{30}$$

$$\mathbf{q}(\xi_0)=-\int_S \mathbf{N}(s)^T\bar{u}_n(\xi_0,s)\xi_0\,\mathrm{d}s, \tag{31}$$

$$\mathbf{E}_0\xi^2\mathbf{a}(\xi)_{,\xi\xi}+(\mathbf{E}_0+\mathbf{E}_1^T-\mathbf{E}_1)\xi\mathbf{a}(\xi)_{,\xi}-\mathbf{E}_2\mathbf{a}(\xi)+\mathbf{F}(\xi)=0, \tag{32}$$

where

$$\mathbf{q}(\xi)=\mathbf{E}_0\xi\mathbf{a}(\xi)_{,\xi}+\mathbf{E}_1^T\mathbf{a}(\xi), \tag{33}$$

$$\mathbf{F}(\xi)=\xi\mathbf{F}_s(\xi)-\xi^2\mathbf{F}_b(\xi), \tag{34}$$

$$\mathbf{B}_1(s)=\mathbf{b}_1(s)\mathbf{N}(s), \tag{35}$$

$$\mathbf{B}_2(s)=\mathbf{b}_2(s)\mathbf{N}(s)_{,s}, \tag{36}$$

$$\mathbf{E}_0=\int_S \mathbf{B}_1(s)^T\mathbf{B}_1(s)|J|\mathrm{d}s, \tag{37}$$

$$\mathbf{E}_1=\int_S \mathbf{B}_2(s)^T\mathbf{B}_1(s)|J|\mathrm{d}s, \tag{38}$$

$$\mathbf{E}_2=\int_S \mathbf{B}_2(s)^T\mathbf{B}_2(s)|J|\mathrm{d}s, \tag{39}$$

$$\mathbf{F}_b(\xi)=\int_S \mathbf{N}(s)^T R|J|\mathrm{d}s, \tag{40}$$

and

$$\mathbf{F}_s(\xi)=\mathbf{N}(s_0)^T\bar{u}_n(\xi,s_0)|J(s_0)|+\mathbf{N}(s_1)^T\bar{u}_n(\xi,s_1)|J(s_1)|. \tag{41}$$

Eq. (32) is the so-called scaled boundary finite-element equation. By introducing the shape function, the Poisson equation has been weakened in the circumferential direction in a finite element manner, but remains strong in the radial direction. Thus, the governing partial differential equation is transformed to an ordinary matrix differential equation in the radial direction. The rank of matrices $\mathbf{E}_0,\mathbf{E}_1,\mathbf{E}_2$ and vectors $\mathbf{a}(\xi),\mathbf{F}(\xi),\mathbf{F}_b(\xi),\mathbf{F}_s(\xi)$ is N (where N is the number of nodes in the boundary S). The solution technique of the ordinary matrix differential equation is described in Appendix A.

Thus, when a proper initial guess u_0 is chosen, we can gradually solve the mth-order deformation equations by means of SBFEM as m increases. It must be noted that the right-hand side term δ_m of the mth-order deformation Eq. (12) becomes more and more complicated as m increases. It means that the non-homogeneous term $\mathbf{F}(\xi)$ will become more and more complicated. This will result in a significant reduction in computational efficiency. To overcome this drawback, Chebyshev interpolation is employed to obtain a polynomial approximation $\mathbf{P}_n(\xi)$ of degree n to $\mathbf{F}(\xi)$, where n is the highest order of Chebyshev polynomials. Replacing $\mathbf{F}(\xi)$ with $\mathbf{P}_n(\xi)$, we can solve the scaled boundary finite element equations with high efficiency. For details of Chebyshev interpolation, please refer to [20].

2.4. Optimal c_0

The Mth-order approximation of $u(x,y)$ is given by

$$\tilde{u}_M(x,y) = u_0(x,y) + \sum_{m=1}^{M} u_m(x,y). \tag{42}$$

Note that, $\tilde{u}_M(x,y)$ is not only dependent upon x and y, but also upon the parameter c_0. Recently, some researches [19,21] have been done to find out the optimal convergence-control parameter c_0 to get a faster convergent series solution.

At the Mth-order approximation, the exact square residual error is defined as follow:

$$\Delta_M = \int_{\Omega} (\mathcal{N}[\tilde{u}_M(x,y)])^2 \,\mathrm{d}\Omega = \int_{\xi_0}^{\xi_1} \int_{s_0}^{s_1} (\mathcal{N}[\tilde{u}_M(\xi,s)])^2 \xi |J| \mathrm{d}s \,\mathrm{d}\xi = \int_{\xi_0}^{\xi_1} \int_{s_0}^{s_1} Ef(\xi,s) \mathrm{d}s \,\mathrm{d}\xi, \tag{43}$$

where

$$Ef(\xi,s) = (\mathcal{N}[\tilde{u}_M(\xi,s)])^2 \xi |J|. \tag{44}$$

However, a large amount of CPU time is needed to calculate the exact residual error Δ_M, especially for high order of approximation. Thus, to decrease the CPU time, the so-called averaged residual error is defined by

$$E_M = \frac{1}{I \cdot J} \sum_{i=0}^{I} \sum_{j=0}^{J} Ef(\xi_0 + i\Delta\xi, s_0 + j\Delta s), \tag{45}$$

where $\Delta\xi = (\xi_1 - \xi_0)/I$ and $\Delta s = (s_1 - s_0)/J$.

E_M is the function of the unknown convergence-control parameter c_0. Obviously, the more quickly E_M decreases to zero, the faster the corresponding homotopy-series solution converges. For a given M, the corresponding optimal values of the convergence-control parameter c_0 are given by the minimum of E_M, corresponding to an algebraic equation

$$\frac{\mathrm{d}E_M}{\mathrm{d}c_0} = 0. \tag{46}$$

3. Numerical examples

To demonstrate the validity of the homotopy-based scaled boundary FEM, assume that the solution of Eq. (5) is exactly equal to a prescribed function $\bar{u}(x,y)$. And then the non-homogeneous term $f(x,y)$ can be determined by substituting the exact solution into Eq. (5). Meanwhile, the functions $g_1(x,y)$ and $g_2(x,y)$ can be set according to $\bar{u}(x,y)$. In this way, we can compare our numerical results with the known exact solution. All of our calculations were done on a laptop PC with a 2.49 GHz CPU and 2 GB RAM.

3.1. For a bounded domain

First of all, we consider the nonlinear 2D Poisson equation (5) defined in a unit bounded domain $\Omega = \{(x,y) : x^2 + y^2 \leqslant 1\}$ with the Dirichlet-type boundary condition $u(x,y) = \bar{g}(x,y)$ on its boundary Γ. Let $\bar{u}(x,y) = x^2 + y^2$ be the solution of Eq. (5), then we have $f(x,y) = 4 + 8(x^2 + y^2)$ and $\bar{g}(x,y) = 1$.

Due to the symmetry, only one quarter of this problem is modeled, as illustrated in Fig. 2. The boundary condition remains the same as $u(x,y) = g_1(x,y) = 1$ on Γ_1, but the boundary condition $u_{,n} = g_2(x,y) = 0$ is enforced on Γ_2 due to the symmetry, where the comma in the subscript designates partial derivative with respect to the variable following the comma. The initial guess is chosen as $u_0(x,y) = \sqrt{x^2 + y^2}$, which satisfies the boundary condition automatically. The scaling centre is chosen at the original of the Cartesian coordinates, and then the boundary Γ_2 is the so-called side-faces which need not to be discretized. The boundary Γ_1 is discretized with three-noded quadratic elements. For a quarter unite circle, we have

$$x_s(s) = \cos(s), \quad y_s(s) = \sin(s), \tag{47}$$

where s is in the range of $[0, \pi/2]$. One kind of mesh is constructed by binary subdivision of the boundary Γ_1, i.e. the mesh consists of two scaled boundary elements, as shown in Fig. 2. To calculate the averaged residual error E_M (45), we use $I = 20$ and $J = 20$.

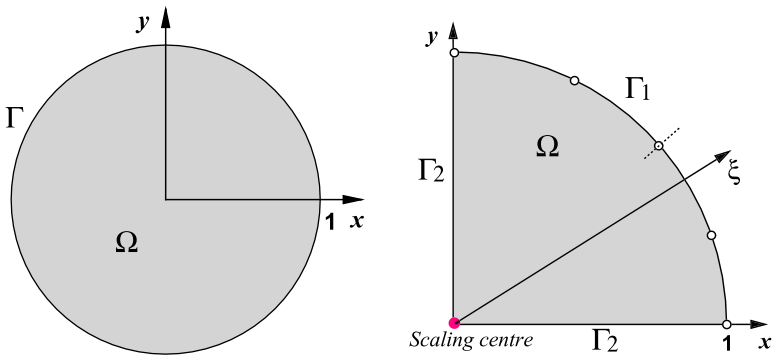

Fig. 2. Bounded domain.

Firstly, let n, the highest order of Chebyshev polynomials, equal to 10. The curves of $\log(\sqrt{E_M})$ versus c_0 at different order of approximation $M = 4,\ 8,\ 12,\ 16$ and 20 are shown in Fig. 3, which indicate that the optimal value of c_0 is about -0.60. In case of $n = 20$, the optimal value of c_0 is close to -0.63, as shown in Table 1. For different values of n, the optimal value of c_0 changes very little. As long as the choice of c_0 is near the optimal value, the homotopy-series solution could converge faster.

For the sake of comparison, c_0 is set as -0.6. The variations of the square root of E_m and the CPU time with the order of approximation M at different $n = 10, 15, 20, 25$ are plotted in Figs. 4 and 5, respectively. Fig. 4 illustrates that, for a given n, the value of E_M decreases as M increases, but when M increases to a certain value M_p, E_M is almost the same. The lowest value of E_M is denoted as E_M^*, which depends on n. The larger value of n, the smaller the value of E_M^*. In Fig. 5, for a given order of approximation M, more CPU time is needed as n increases. It means that the highest order of Chebyshev polynomials affect not only the accuracy but also the computational efficiency.

As shown in Figs. 4 and 5, in the case of $n = 25$ and $M = 100$, the CPU time is about 9236 seconds with $\sqrt{E_M} = 3.6 \times 10^{-17}$; in the case of $n = 10$ and $M = 20$, $\sqrt{E_M}$ has decreased to 1.1×10^{-5}, and the CPU time is about 137 s, less than 3 min.

In the frame of the SBFEM, properties of the standard finite-element method are maintained in the circumferential direction, but the analytical solution is provided in the radial direction. As long as the high precision is ensured in the radial direction and the number of surface elements is large enough in the circumferential direction, we could guarantee the accuracy of the numerical solution in the whole domain. In this example, since the boundary S is a quarter of circle and the exact solution on any path from the scaling centre to the boundary is the same, the impact from the discretization of the boundary can be ignored. So we only need to compare the numerical approximations with the exact solution in a given radial path. As shown in Figs. 6 and 7, the higher the order of approximation, the less the absolute error between the numerical approximation and the exact solution.

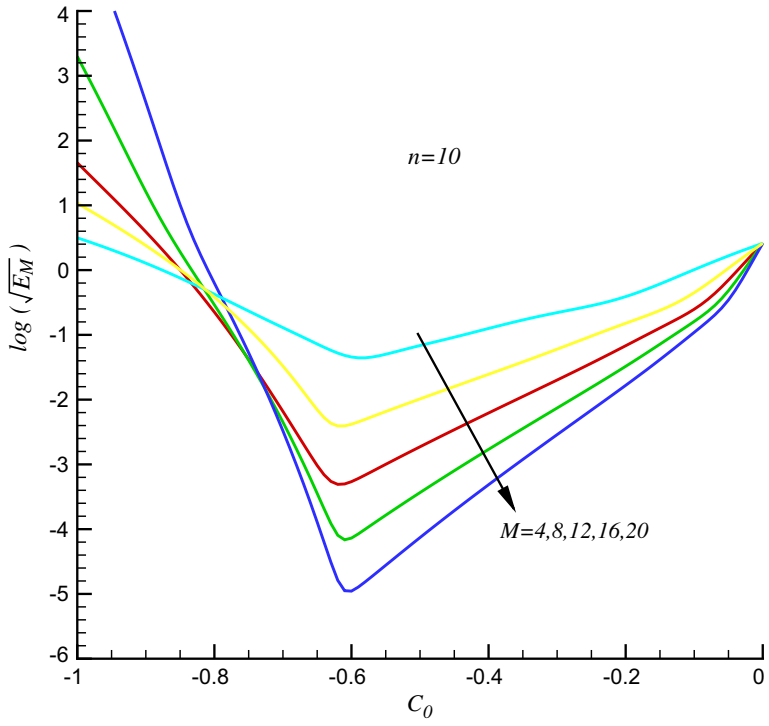

Fig. 3. The logarithm of $\sqrt{E_M}$ versus c_0 in the case of $n = 10$ for the bounded domain.

Table 1
Optimal value of c_0 in the case of $n = 20$ for the bounded domain.

Order of approximation M	Optimal value of c_0	Minimum value of $\sqrt{E_M}$
4	−0.585	4.40E − 02
6	−0.604	1.24E − 02
8	−0.616	3.90E − 03
10	−0.620	1.29E − 03
12	−0.623	4.39E − 04
14	−0.625	1.50E − 04
16	−0.627	5.23E − 05
18	−0.628	1.84E − 05
20	−0.630	6.52E − 06

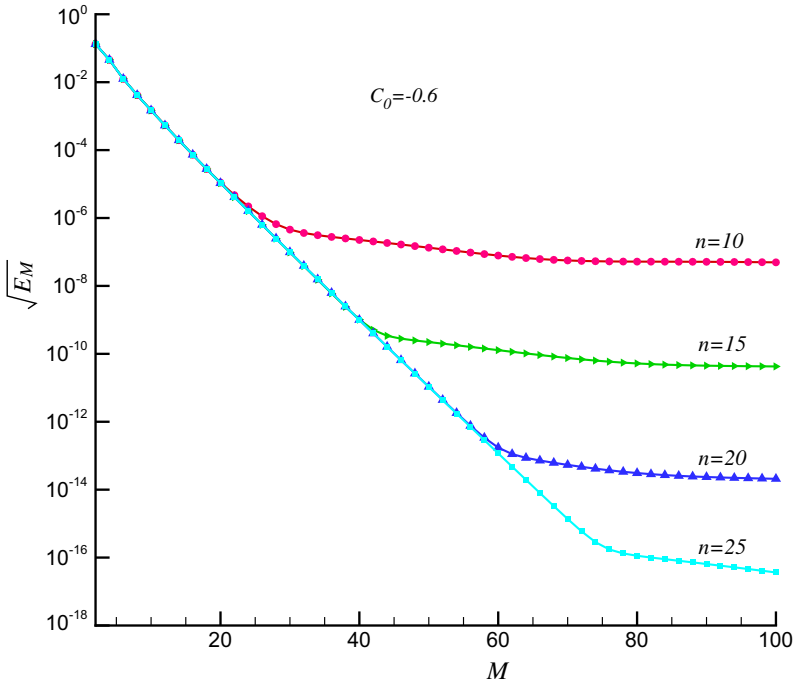

Fig. 4. The square root of E_M versus M in the case of $c_0 = -0.6$ for the bounded domain.

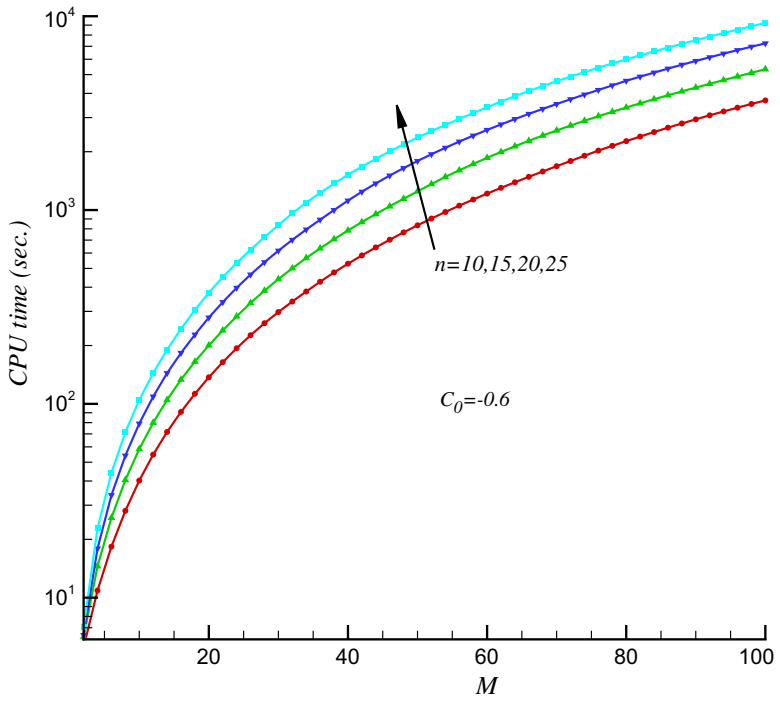

Fig. 5. The CPU time versus M in the case of $c_0 = -0.6$ for the bounded domain.

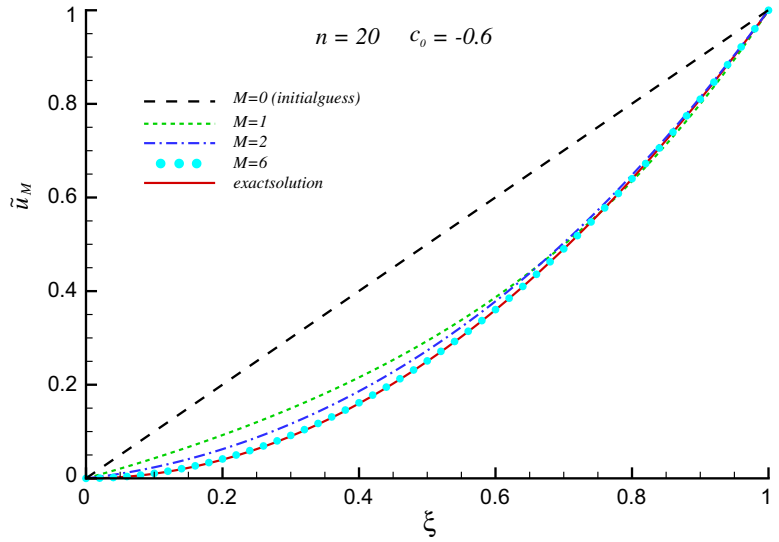

Fig. 6. The approximation $\tilde{u}_M$ versus ξ in the case of $n = 20$ and $c_0 = -0.6$ for the bounded domain.

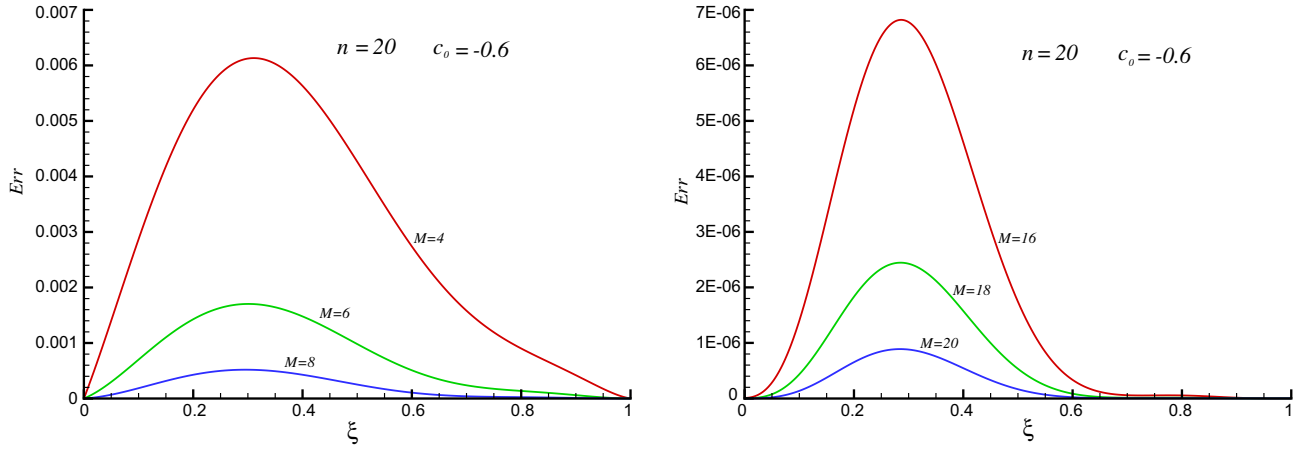

Fig. 7. The error ($|\tilde{u}_M - u_{exact}|$) versus ξ in the case of $n = 20$ and $c_0 = -0.6$ for the bounded domain.

3.2. For an unbounded domain

Secondly, the nonlinear 2D Poisson equation (5) defined in an unbounded domain $\Omega = \{(x,y) : x^2 + y^2 \geqslant 1\}$ with the Dirichlet-type boundary is considered. Let the exact solution $\bar{u}(x,y)$ equal to $1/(x^2+y^2)$, then we have $f(x,y) = 4(x^2+y^2+2)/(x^2+y^2)^3$ and $g(x,y) = \bar{g}(x,y) = 1$.

Similarly, due to the symmetry, only one quarter of this problem is modeled, as illustrated in Fig. 8. The boundary condition remains the same as $u(x,y) = g_1(x,y) = 1$ on Γ_1, and the condition $u_{,n} = g_2(x,y) = 0$ is enforced on Γ_2 due to the sym-

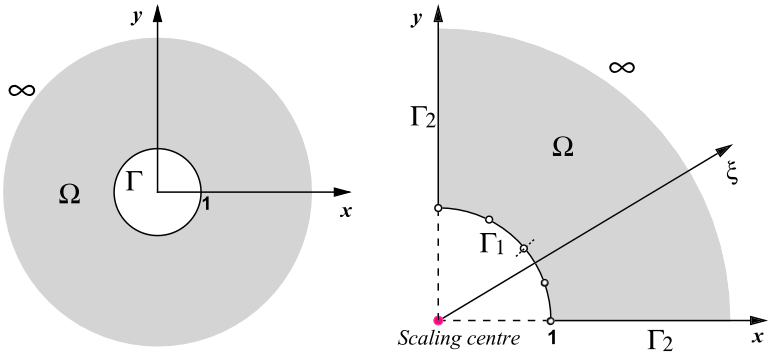

Fig. 8. Unbounded domain.

metry. The initial guess is chosen as $u_0(x,y) = 1/\sqrt{x^2+y^2}$. Since the boundary S is a quarter of unit circle, the scaled boundary transformation is the same as the bounded domain, but the scaling factor ξ varies from 1 to $+\infty$. In order to calculate E_M, we use $\Delta\xi = 0.1$, $I = 100$ and $J = 20$ in Eq. (45). So the optimal convergence-control parameter c_0 can be calculated by Eq. (46).

As shown in Table 2, in the case of $n = 10$, the optimal value of c_0 is about -0.59. Fig. 9 shows that if c_0 is chosen near the optimal value, E_M decreases faster as the order of approximation M increases.

The value of the highest order of Chebyshev polynomials has a great influence on the computational efficiency and accuracy, as shown in Figs. 10 and 11. In the case of $n = 20$ and $c_0 = -0.6$, as M increases to 20, the square root of the averaged residual error E_M decreases to 8.6×10^{-7}, but the total CPU time is only about 142 s. Similarly, let n equal to 20 and the convergence-control parameter c_0 equal to -0.6, then we compare the numerical approximations with the exact solution in a given radial path. The exact solution $u(\xi, s)$ is equal to $1/\xi^2$, which tends to zero as $\xi \to \infty$. Figs. 12 and 13 illustrate that the absolute error between the numerical approximation and the exact solution decreases quickly as M increases.

4. Conclusions

A new approach has been proposed for 2D nonlinear Poisson-type problems by combining the scaled boundary finite-element method with the homotopy analysis method. In this approach, the original nonlinear problem is transferred into an infinite number of linear Poisson-type sub-problems by the HAM, and then all those linear Poisson-type sub-problems are solved by means of the SBFEM. Numerical examples governed by a 2D nonlinear Poisson equation, including a bounded

Table 2
Optimal value of c_0 in the case of $n = 10$ for the unbounded domain.

Order of approximation M	Optimal value of c_0	Minimum value of $\sqrt{E_M}$
4	−0.592	2.04E − 03
6	−0.599	7.30E − 04
8	−0.602	2.55E − 04
10	−0.603	8.28E − 05
12	−0.600	2.79E − 05
14	−0.595	1.04E − 05
16	−0.592	4.14E − 06
18	−0.590	1.72E − 06
20	−0.589	7.57E − 07

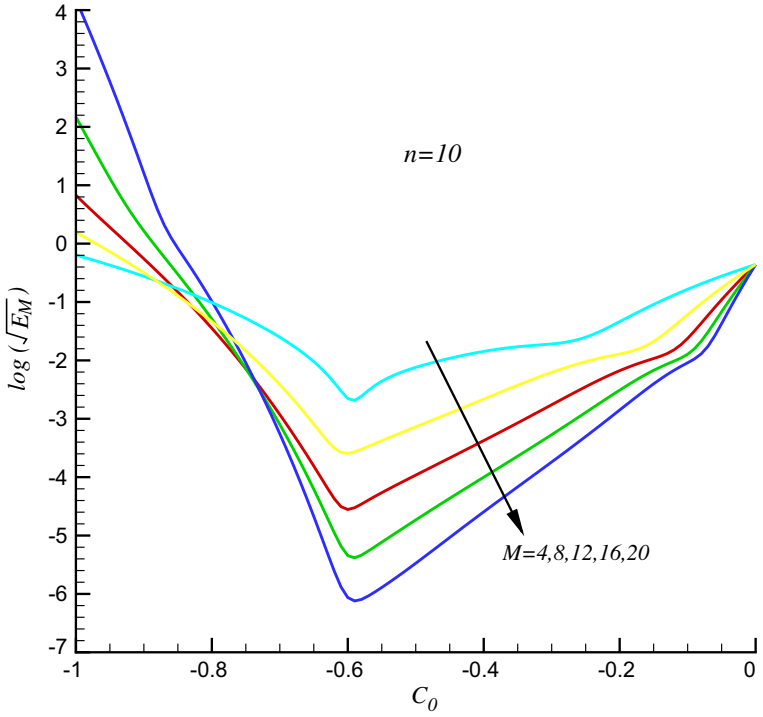

Fig. 9. The logarithm of $\sqrt{E_M}$ versus c_0 in the case of $n = 10$ for the unbounded domain.

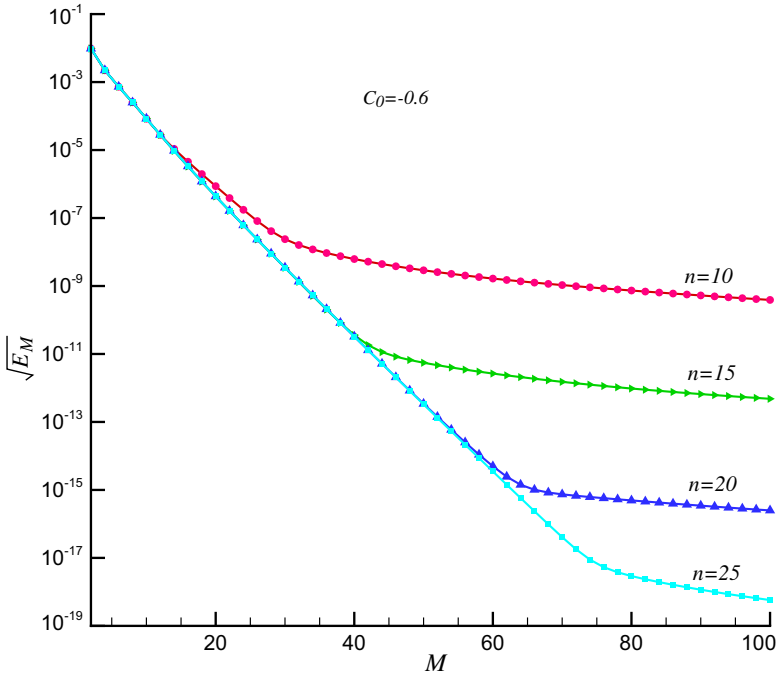

Fig. 10. The square root of E_M versus M in the case of $c_0 = -0.6$ for the unbounded domain.

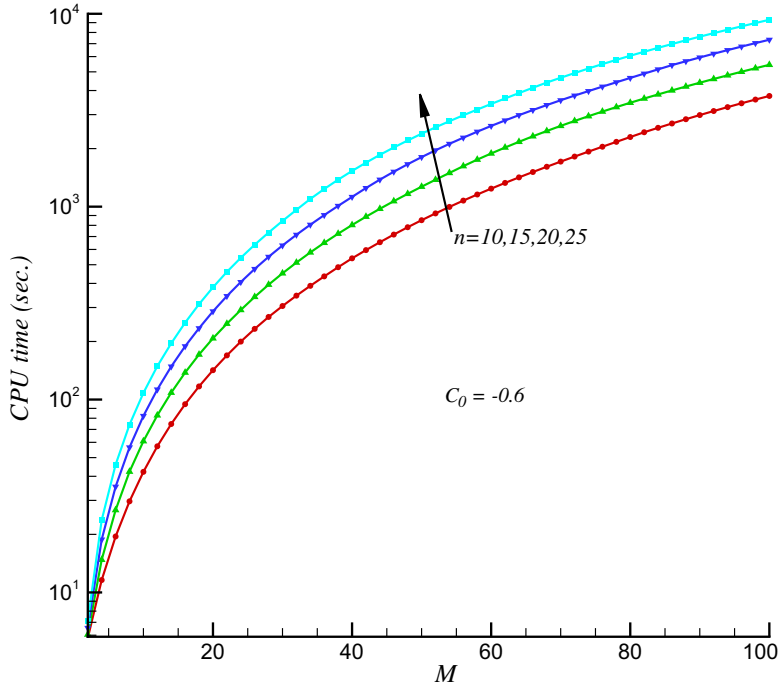

Fig. 11. The CPU time versus M in the case of $c_0 = -0.6$ for the unbounded domain.

domain and an unbounded domain, are solved by this new approach, respectively. All the numerical results demonstrate the high efficiency and accuracy of our approach.

Note that the HAM enjoys great freedom in choosing initial guess u_0. If the convergence-control parameter c_0 is selected properly, the Mth-order approximation $\tilde{u}_M$ is better than the free selected initial approximation. So, it can provide us with a family of iterative formulae.

Clearly, this new approach has great potential to the nonlinear problems. In this paper, we have successfully extended the SBFEM to solve nonlinear problems. Meanwhile, the idea of the HAM has been further applied to numerical calculation. Though more research effort is needed to develop and improve this new approach, it may not be far away to apply our approach to solve more nonlinear engineering problems with increased complexity.

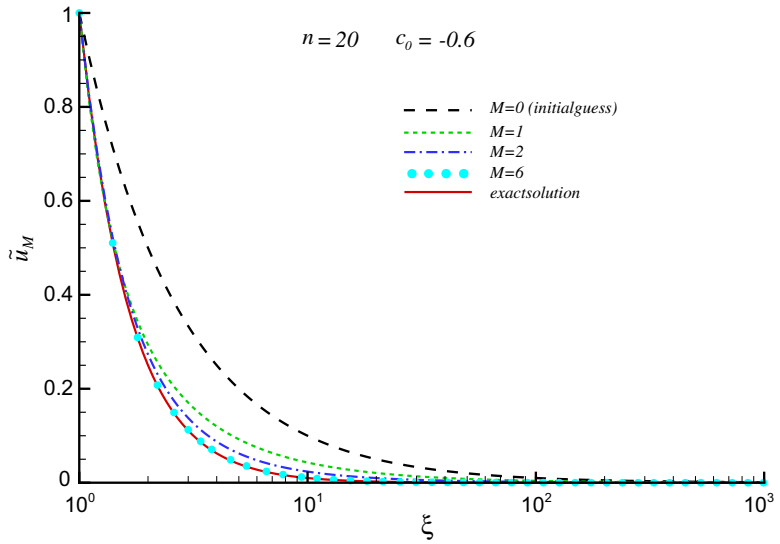

Fig. 12. The approximation $\tilde{u}_M$ versus ξ in the case of $n = 20$ and $c_0 = -0.6$ for the unbounded domain.

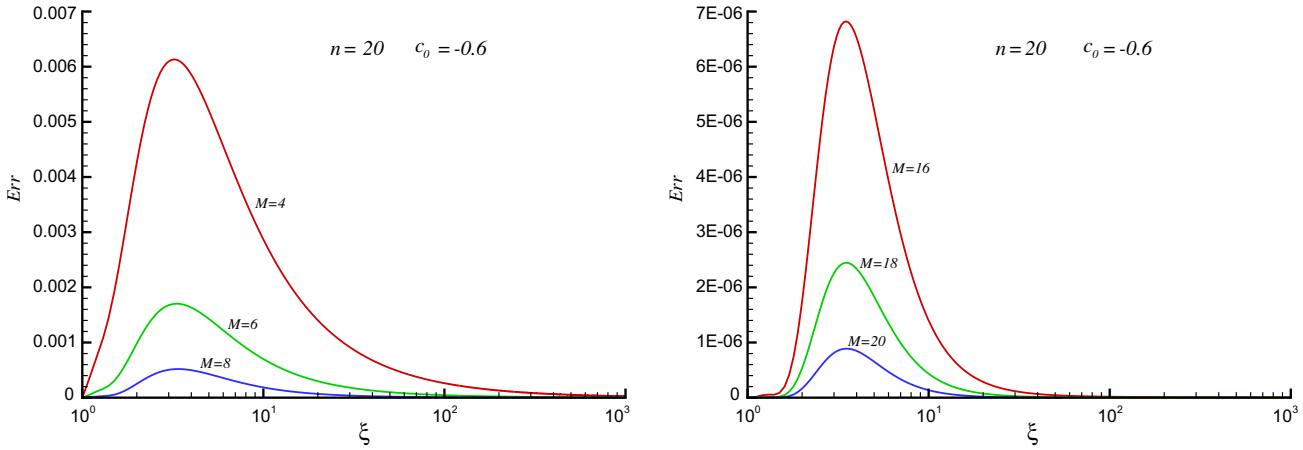

Fig. 13. The error ($|\tilde{u}_M - u_{exact}|$) versus ξ in the case of $n = 20$ and $c_0 = -0.6$ for the unbounded domain.

Acknowledgements

The work is partly supported by National Natural Science Foundation of China (Approval No. 10872129) and State Key Lab of Ocean Engineering (Approval No. GKZD010002). Thanks to Prof. Tao (Newcastle University) for the suggestive discussions and valuable comments.

Appendix A. Solution technique of SBFEM

By introducing the variable

$$\mathbf{X}(\xi) = \begin{Bmatrix} \mathbf{a}(\xi) \\ \mathbf{q}(\xi) \end{Bmatrix}, \tag{A.1}$$

Eq. (32) is transformed into the first-order ordinary differential equations

$$\xi \mathbf{X}(\xi)_{,\xi} = -\mathbf{Z}\mathbf{X}(\xi) - \begin{Bmatrix} \mathbf{0} \\ \mathbf{F}(\xi) \end{Bmatrix} \tag{A.2}$$

with a Hamiltonian matrix

$$\mathbf{Z} = \begin{bmatrix} \mathbf{E}_0^{-1}\mathbf{E}_1^T & -\mathbf{E}_0^{-1} \\ -\mathbf{E}_2 + \mathbf{E}_1\mathbf{E}_0^{-1}\mathbf{E}_1^T & -\mathbf{E}_1\mathbf{E}_0^{-1} \end{bmatrix}. \tag{A.3}$$

The eigenvalue problem is formulated as

$$\mathbf{Z}\boldsymbol{\Phi} = -\boldsymbol{\Phi}\boldsymbol{\Lambda} \tag{A.4}$$

with the eigenvalues of the Hamiltonian matrix

$$\boldsymbol{\Lambda} = \begin{bmatrix} -\boldsymbol{\Lambda}_0 & \\ & \boldsymbol{\Lambda}_0 \end{bmatrix}, \quad \boldsymbol{\Lambda}_0 = \lceil \lambda_i \rfloor \tag{A.5}$$

and the eigenvector matrix partitioned conformably

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}_{11} & \boldsymbol{\Phi}_{12} \\ \boldsymbol{\Phi}_{21} & \boldsymbol{\Phi}_{22} \end{bmatrix}. \tag{A.6}$$

The real parts of all elements λ_i are non-positive. The solution of Eq. (A.2) equals

$$\mathbf{X}(\xi) = \begin{bmatrix} \boldsymbol{\Phi}_{11} & \boldsymbol{\Phi}_{12} \\ \boldsymbol{\Phi}_{21} & \boldsymbol{\Phi}_{22} \end{bmatrix} \begin{bmatrix} \xi^{-\boldsymbol{\Lambda}_0} & \\ & \xi^{\boldsymbol{\Lambda}_0} \end{bmatrix} \begin{Bmatrix} \mathbf{C}_1(\xi) \\ \mathbf{C}_2(\xi) \end{Bmatrix}, \tag{A.7}$$

where

$$\mathbf{C}_1(\xi) = \mathbf{c}_1 - \int_1^{\xi} \tau^{-1}\tau^{\boldsymbol{\Lambda}_0}\mathbf{A}_{12}\mathbf{F}(\tau)\mathrm{d}\tau, \tag{A.8}$$

$$\mathbf{C}_2(\xi) = \mathbf{c}_2 - \int_1^{\xi} \tau^{-1}\tau^{-\boldsymbol{\Lambda}_0}\mathbf{A}_{22}\mathbf{F}(\tau)\mathrm{d}\tau, \tag{A.9}$$

$\mathbf{c}_1, \mathbf{c}_1$ are integration constants and

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} = \boldsymbol{\Phi}^{-1}. \tag{A.10}$$

Substituting (A.7) into (A.1) yields

$$\mathbf{a}(\xi) = \boldsymbol{\Phi}_{11}\xi^{-\boldsymbol{\Lambda}_0}\mathbf{C}_1(\xi) + \boldsymbol{\Phi}_{12}\xi^{\boldsymbol{\Lambda}_0}\mathbf{C}_2(\xi) \tag{A.11}$$

$$\mathbf{q}(\xi) = \boldsymbol{\Phi}_{21}\xi^{-\boldsymbol{\Lambda}_0}\mathbf{C}_1(\xi) + \boldsymbol{\Phi}_{22}\xi^{\boldsymbol{\Lambda}_0}\mathbf{C}_2(\xi) \tag{A.12}$$

For a bounded domain, $\mathbf{a}(\xi)$ at $\xi = 0$ must remain finite, leading to $\mathbf{C}_2(\xi)|_{\xi=0} = 0$, then

$$\mathbf{c}_2 = \int_1^{0} \tau^{-1}\tau^{-\boldsymbol{\Lambda}_0}\mathbf{A}_{22}\mathbf{F}(\tau)\mathrm{d}\tau. \tag{A.13}$$

Substituting (A.13) into (A.9) yields

$$\mathbf{C}_2(\xi) = -\int_0^{\xi} \tau^{-1}\tau^{-\boldsymbol{\Lambda}_0}\mathbf{A}_{22}\mathbf{F}(\tau)\mathrm{d}\tau. \tag{A.14}$$

And then substituting Eqs. (A.8) and (A.14) into Eqs. (A.11) and (A.12) results in

$$\mathbf{a}(\xi) = \boldsymbol{\Phi}_{11}\xi^{-\boldsymbol{\Lambda}_0}\left(\mathbf{c}_1 + \int_{\xi}^{1} \tau^{-1}\tau^{\boldsymbol{\Lambda}_0}\mathbf{A}_{12}\mathbf{F}(\tau)\mathrm{d}\tau\right) - \boldsymbol{\Phi}_{12}\xi^{\boldsymbol{\Lambda}_0}\int_0^{\xi} \tau^{-1}\tau^{-\boldsymbol{\Lambda}_0}\mathbf{A}_{22}\mathbf{F}(\tau)\mathrm{d}\tau \tag{A.15}$$

and

$$\mathbf{q}(\xi) = \boldsymbol{\Phi}_{21}\xi^{-\boldsymbol{\Lambda}_0}\left(\mathbf{c}_1 + \int_{\xi}^{1} \tau^{-1}\tau^{\boldsymbol{\Lambda}_0}\mathbf{A}_{12}\mathbf{F}(\tau)\mathrm{d}\tau\right) - \boldsymbol{\Phi}_{22}\xi^{\boldsymbol{\Lambda}_0}\int_0^{\xi} \tau^{-1}\tau^{-\boldsymbol{\Lambda}_0}\mathbf{A}_{22}\mathbf{F}(\tau)\mathrm{d}\tau. \tag{A.16}$$

Note that $\mathbf{c}_1$ is determined by the boundary condition

$$\mathbf{a}(\xi)|_{\xi=1} = \boldsymbol{\Phi}_{11}\mathbf{c}_1 - \boldsymbol{\Phi}_{12}\int_0^{1} \tau^{-1}\tau^{-\boldsymbol{\Lambda}_0}\mathbf{A}_{22}\mathbf{F}(\tau)\mathrm{d}\tau \tag{A.17}$$

or

$$\mathbf{q}(\xi)|_{\xi=1} = \boldsymbol{\Phi}_{21}\mathbf{c}_1 - \boldsymbol{\Phi}_{22}\int_0^{1} \tau^{-1}\tau^{-\boldsymbol{\Lambda}_0}\mathbf{A}_{22}\mathbf{F}(\tau)\mathrm{d}\tau. \tag{A.18}$$

For an unbounded domain, $\mathbf{a}(\xi)$ as $\xi \to \infty$ must remain finite, leading to $\mathbf{C}_1(\xi)|_{\xi\to\infty} = 0$. The solution procedure is similar, please refer to [3].

References

[1] Wolf JP, Song C. The scaled boundary finite-element method – a primer: derivation. Comput Struct 2000;78:191–210.
[2] Wolf JP, Song C. The scaled boundary finite-element method – a fundamental solution-less boundary-element method. Comput Meth Appl Mech Eng 2001;190:5551–68.
[3] Wolf JP. The scaled boundary finite-element method. Chichester (UK): Wiley; 2003.
[4] Song C, Wolf JP. The scaled boundary finite-element method – alias consistent infinitesimal finite-element cell method – for elastodynamics. Comput Meth Appl Mech Eng 1997;147:329–55.
[5] Deeks AJ, Cheng L. Potential flow around obstacles using the scaled boundary finite-element method. Int J Numer Meth Fluids 2003;41:721–41.
[6] Tao L, Song H, Chakrabarti S. Scaled boundary FEM solution of short-crested wave diffraction by a vertical cylinder. Comput Meth Appl Mech Eng 2007;197(1–4):232–42.
[7] Song H, Tao L, Chakrabarti S. Modelling of water wave interaction with multiple cylinders of arbitrary shape. J Comput Phys 2010;229(5):1498–513.
[8] Song H, Tao L. An efficient scaled boundary FEM model for wave interaction with a nonuniform porous cylinder. Int J Numer Meth Fluids. doi:10.1002/fld.2080.
[9] Doherty JP, Deeks AJ. Adaptive coupling of the finite-element and scaled boundary finite-element methods for non-linear analysis of unbounded media. Comput Geotech 2005;32:436–44.
[10] Deeks AJ, Augarde CE. A hybrid meshless Petrov–Galerkin method for unbounded domains. Comput Meth Appl Mech Eng 2007;196:843–52.
[11] Chidgzey SR, Trevelyan J, Deeks AJ. Coupling of the boundary element method and the scaled boundary finite element method for computations in fracture mechanics. Comput Struct 2008;86:1198–203.
[12] Liao SJ. The proposed homotopy analysis technique for the solution of nonlinear problems. Ph.D. Thesis, Shanghai Jiao Tong University; 1992.
[13] Liao SJ. A kind of approximate solution technique which does not depend upon small parameters (II): an application in fluid mechanics. Int J Nonlinear Mech 1997;32:815–22.
[14] Liao SJ. An explicit, totally analytic approximation of Blasius viscous flow problems. Int J Nonlinear Mech 1999;34(4):759–78.
[15] Liao SJ. Beyond perturbation: introduction to the homotopy analysis method. Boca Raton (FL): Chapman & Hall; 2003.
[16] Liao SJ. On the homotopy analysis method for nonlinear problems. Appl Math Comput 2004;147:499–513.
[17] Liao SJ, Tan Y. A general approach to obtain series solutions of nonlinear differential equations. Stud Appl Math 2007;119:297–355.
[18] Liao SJ. Notes on the homotopy analysis method: some definitions and theorems. Commun Nonlinear Sci Numer Simulat 2009;14:983–97.
[19] Liao SJ. An optimal homotopy-analysis approach for strongly nonlinear differential equations. Commun Nonlinear Sci Numer Simulat. doi:10.1016/j.cnsns.2009.09.002.
[20] Mason JC, Handscomb DC. Chebyshev polynomials. Boca Raton (FL): Chapman & Hall; 2003.
[21] Niu Z, Wang Ch. A one-step optimal homotopy analysis method for nonlinear differential equations. Commun Nonlinear Sci Numer Simulat. doi:10.1016/j.cnsns.2009.08.014.