 ICCES, vol.3, no.3, pp.127-132, 2007

Solving Partial Differential Equations With Point Collocation And One-Dimensional Integrated Interpolation Schemes

N. Mai-Duy[1] and T. Tran-Cong[1]

Summary

This lecture presents an overview of the Integral Collocation formulation for numerically solving partial differential equations (PDEs). However, due to space limitation, the paper only describes the latest development, namely schemes based only on one-dimensional (1D) integrated interpolation even in multi-dimensional problems. The proposed technique is examined with Chebyshev polynomials and radial basis functions (RBFs). The latter can be used in both regular and irregular domains. For both basis functions, the accuracy and convergence rates of the new technique are better than those of the differential formulation.

Introduction

Point collocation is the simplest way to discretize PDEs (e.g. no mesh and no integration associated with this process are required). However, in general, this approach is seen not to be as stable as those associated with the weak form and the inverse statement.

For the conventional collocation techniques, the construction of the approximations for the field variables is based on differentiation. For example, an approximate solution can be sought in RBF or truncated Chebyshev series forms which are then differentiated to obtain expressions for derivative functions. It was proved that there is a reduction in accuracy for derivative functions [1],[2].

To enhance the stability and accuracy of a collocation scheme, it was proposed in [3],[4] that the expressions for the variables are constructed based on integration (the integral formulation). In this paper, numerical collocation schemes, which are based on 1D integrated RBF and Chebyshev polynomial schemes, are considered and presented in detail through fourth-order PDEs, which have double boundary conditions.

One-dimensional integrated interpolation schemes

Consider the biharmonic equation

$$\nabla^4 u = b(x,y), \tag{1}$$

subject to Dirichlet boundary conditions u and $\partial u/\partial n$ ($b-$a driving function and $n-$the direction normal to the local boundary). For brevity, only the case of rectangular domains is detailed here. The problem domain can be discretized by using a

[1]Computational Engineering and Science Research Centre (CESRC), Faculty of Engineering and Surveying, The University of Southern Queensland, QLD 4350, Australia

uniform/nonuniform Cartesian grid for RBFs and a tensor product grid formed by the Gauss-Lobatto points for Chebyshev polynomials.

The variable u and its derivatives along a grid line that run parallel to the $x-$axis can be approximated by

$$\frac{\partial^4 u(x)}{\partial x^4} = \sum_{i=1}^{N_x} w^{(i)} H_{[4]}^{(i)}(x), \tag{2}$$

$$\frac{\partial^3 u(x)}{\partial x^3} = \sum_{i=1}^{N_x} w^{(i)} H_{[3]}^{(i)}(x) + c_1, \tag{3}$$

$$\frac{\partial^2 u(x)}{\partial x^2} = \sum_{i=1}^{N_x} w^{(i)} H_{[2]}^{(i)}(x) + c_1 x + c_2, \tag{4}$$

$$\frac{\partial u(x)}{\partial x} = \sum_{i=1}^{N_x} w^{(i)} H_{[1]}^{(i)}(x) + c_1 \frac{x^2}{2} + c_2 x + c_3, \tag{5}$$

$$u(x) = \sum_{i=1}^{N_x} w^{(i)} H_{[0]}^{(i)}(x) + c_1 \frac{x^3}{6} + c_2 \frac{x^2}{2} + c_3 x + c_4, \tag{6}$$

where $H_{[4]}(x)$ is a radial basis function or a Chebyshev polynomial, N_x the number of grid points in the $x-$direction, $H_{[3]}(x) = \int H_{[4]}(x)dx$, $H_{[2]}(x) = \int H_{[3]}(x)dx$, $H_{[1]}(x) = \int H_{[2]}(x)dx$ and $H_{[0]}(x) = \int H_{[1]}(x)dx$, and w unknown weights.

It is more convenient to work in the physical space than in the spectral space. The presence of four integration constants in (6) allows one to add four extra equations to the conversion system. These equations can be chosen to be the governing equation and normal derivative boundary conditions at both ends of the line. The conversion system can thus be given by

$$\begin{pmatrix} \widehat{u} \\ \widehat{f} \end{pmatrix} = \begin{bmatrix} \mathscr{H} \\ \mathscr{K} \end{bmatrix} \begin{pmatrix} \widehat{w} \\ \widehat{c} \end{pmatrix}, \tag{7}$$

where

$$\widehat{w} = \begin{pmatrix} w^{(1)} \\ w^{(2)} \\ \cdots \\ w^{(N_x)} \end{pmatrix}, \qquad \widehat{c} = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix}, \qquad \widehat{u} = \begin{pmatrix} u^{(1)} \\ u^{(2)} \\ \cdots \\ u^{(N_x)} \end{pmatrix},$$

$$\mathscr{H} = \begin{bmatrix} H_{[0]}^{(1)}(x^{(1)}) & \cdots & H_{[0]}^{(N_x)}(x^{(1)}) & x^{(1)3}/6 & x^{(1)2}/2 & x^{(1)} & 1 \\ H_{[0]}^{(1)}(x^{(2)}) & \cdots & H_{[0]}^{(N_x)}(x^{(2)}) & x^{(2)3}/6 & x^{(2)2}/2 & x^{(2)} & 1 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ H_{[0]}^{(1)}(x^{(N_x)}) & \cdots & H_{[0]}^{(N_x)}(x^{(N_x)}) & x^{(N_x)3}/6 & x^{(N_x)2}/2 & x^{(N_x)} & 1 \end{bmatrix},$$

$$\mathscr{K} = \begin{bmatrix} H_{[1]}^{(1)}(x^{(1)}) & \cdots & H_{[1]}^{(N_x)}(x^{(1)}) & x^{(1)2}/2 & x^{(1)} & 1 & 0 \\ H_{[1]}^{(1)}(x^{(N_x)}) & \cdots & H_{[1]}^{(N_x)}(x^{(N_x)}) & x^{(N_x)2}/2 & x^{(N_x)} & 1 & 0 \\ H_{[4]}^{(1)}(x^{(1)}) & \cdots & H_{[4]}^{(N_x)}(x^{(1)}) & 0 & 0 & 0 & 0 \\ H_{[4]}^{(1)}(x^{(N_x)}) & \cdots & H_{[4]}^{(N_x)}(x^{(N_x)}) & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$\widehat{f} = \begin{pmatrix} \frac{\partial u}{\partial x}(x^{(1)}) \\ \frac{\partial u}{\partial x}(x^{(N_x)}) \\ b(x^{(1)}) - 2\frac{\partial^4 u}{\partial x^2 y^2}(x^{(1)}) - \frac{\partial^4 u}{\partial y^4}(x^{(1)}) \\ b(x^{(N_x)}) - 2\frac{\partial^4 u}{\partial x^2 y^2}(x^{(N_x)}) - \frac{\partial^4 u}{\partial y^4}(x^{(N_x)}) \end{pmatrix}.$$

Solving (7) yields

$$\begin{pmatrix} \widehat{w} \\ \widehat{c} \end{pmatrix} = \begin{bmatrix} \mathscr{H} \\ \mathscr{K} \end{bmatrix}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{f} \end{pmatrix} = \mathscr{C}^{-1} \begin{pmatrix} \widehat{u} \\ \widehat{f} \end{pmatrix}, \tag{8}$$

where $\mathscr{C}$ is the conversion matrix.

Substitution of (8) into (2)-(5) yields

$$\frac{\partial^4 u(x)}{\partial x^4} = \left(H_{[4]}^{(1)}(x), H_{[4]}^{(2)}(x), \cdots, H_{[4]}^{(N)}(x), 0,0,0,0\right)\mathscr{C}^{-1}\begin{pmatrix} \widehat{u} \\ \widehat{f} \end{pmatrix}, \tag{9}$$

$$\frac{\partial^3 u(x)}{\partial x^3} = \left(H_{[3]}^{(1)}(x), H_{[3]}^{(2)}(x), \cdots, H_{[3]}^{(N)}(x), 1,0,0,0\right)\mathscr{C}^{-1}\begin{pmatrix} \widehat{u} \\ \widehat{f} \end{pmatrix}, \tag{10}$$

$$\frac{\partial^2 u(x)}{\partial x^2} = \left(H_{[2]}^{(1)}(x), H_{[2]}^{(2)}(x), \cdots, H_{[2]}^{(N)}(x), x,1,0,0\right)\mathscr{C}^{-1}\begin{pmatrix} \widehat{u} \\ \widehat{f} \end{pmatrix}, \tag{11}$$

$$\frac{\partial u(x)}{\partial x} = \left(H_{[1]}^{(1)}(x), H_{[1]}^{(2)}(x), \cdots, H_{[1]}^{(N)}(x), \frac{x^2}{2},x,1,0\right)\mathscr{C}^{-1}\begin{pmatrix} \widehat{u} \\ \widehat{f} \end{pmatrix}. \tag{12}$$

It is noted that normal derivative boundary conditions are imposed in (9)-(12).

The values of the ith-order derivative of u ($i = \{1,2,3,4\}$) at the grid points along a horizontal line can be computed by

$$\widehat{\frac{\partial^i u}{\partial x^i}} = \widehat{\mathscr{D}}_{ix}\begin{pmatrix} \widehat{u} \\ \widehat{f} \end{pmatrix}, \qquad i = \{1,2,3,4\}, \tag{13}$$

where $\widehat{\frac{\partial^i u}{\partial x^i}} = \left(\frac{\partial^i u}{\partial x^i}^{(1)}, \frac{\partial^i u}{\partial x^i}^{(2)}, \cdots, \frac{\partial^i u}{\partial x^i}^{(N_x)}\right)^T$ and $\widehat{\mathscr{D}}_{ix}$ is a known $N_x \times (N_x+4)$ matrix.

Expression (13) can be rewritten as $\widehat{\frac{\partial^i u}{\partial x^i}} = \widehat{\mathscr{D}1}_{ix}\widehat{u} + \widehat{\mathscr{D}2}_{ix}\widehat{f}$, where $\widehat{\mathscr{D}1}_{ix}$ and $\widehat{\mathscr{D}2}_{ix}$ are matrices that are formed by the first N_x columns and the last four columns of the matrix $\widehat{\mathscr{D}}_{ix}$, respectively. The extra information vector $\widehat{f}$ (components f_3 and f_4) contains some unknown values—the mixed partial derivative $\partial^4 u/\partial x^2 \partial y^2$ at the two boundary points. Fortunately, these unknown values can be replaced with linear combinations of nodal values of the variable u (the detailed expression of $\partial^4 u/\partial x^2 \partial y^2$ will be given later on). As a result, one can express (13) in terms of nodal variable values only. The values of the ith-order derivative of u with respect to y along a vertical line will be obtained in the same way.

The approximations for derivatives over 2D grids can be conveniently constructed by means of Kronecker tensor products. Assuming that the grid points are numbered from bottom to top and from left to right, the values of derivatives of u at the grid points can be computed by

$$\widetilde{\frac{\partial^i u}{\partial x^i}} = \left(\widehat{\mathscr{D}}_{ix} \otimes \mathscr{I}_y\right) \widetilde{u} + \widetilde{k}_{ix}, \qquad \widetilde{\frac{\partial^i u}{\partial y^i}} = \left(\mathscr{I}_x \otimes \widehat{\mathscr{D}}_{iy}\right) \widetilde{u} + \widetilde{k}_{iy}, \tag{14}$$

where $\mathscr{I}_x$ and $\mathscr{I}_y$ are the identity matrices of dimension $N_x \times N_x$ and $N_y \times N_y$, respectively, $\widetilde{k}_{ix}$ and $\widetilde{k}_{iy}$ are known vectors and and $\widetilde{u} = \left(u^{(1)}, u^{(2)}, \cdots, u^{(N_x N_y)}\right)^T$. The mixed fourth-order partial derivative can be computed according to the following relation

$$\frac{\partial^4 u}{\partial^2 x \partial^2 y} = \frac{1}{2}\left[\frac{\partial^2}{\partial x^2}\left(\frac{\partial^2 u}{\partial y^2}\right) + \frac{\partial^2}{\partial y^2}\left(\frac{\partial^2 u}{\partial x^2}\right)\right], \tag{15}$$

where relevant second-order derivatives are decomposed into RBFs or Chebyshev polynomials with the extra information being the values of the first-order derivatives at the boundary points. In the case of irregular domains, the extra information vector $\widehat{f}$ in (7) and (15) needs to be modified and the assembly process is similar to that of finite element methods. Further details will be presented at the meeting.

Numerical examples

In the case of RBFs, the width of the ith RBF $(a^{(i)})$ is simply chosen to be the minimum distance from the ith centre to neighbouring centres. When compared with the case of using 2D integrated RBF interpolation schemes, the present technique requires much less computational work because the construction of RBF approximations involves only N_x or N_y points instead of the total number of points. As a result, larger numbers of nodes can be employed (e.g., up to about 6400 nodes employed here). In addition to the results shown below, further examples will be presented at the meeting.

Example 1 Consider a square domain $-1 \leq x, y \leq 1$. The driving function and the exact solution are given by

$$b(x,y) = 16(\pi^2 - 1)^2 \left[\sin(2\pi x)\cosh(2y) - \cos(2\pi x)\sinh(2y)\right], \tag{16}$$

$$u^{(e)}(x,y) = \sin(2\pi x)\cosh(2y) - \cos(2\pi x)\sinh(2y). \tag{17}$$

Table 1: Example 1, Chebyshev polynomials. A comparison of accuracy and convergence speed between the differential (DF) and integral (IF) formulations. Both formulations use the same discretizations and they result in the systems of algebraic equations of the same number of unknowns. The order of accuracy is measured for the first eight sets ($N = N_x - 1 = N_y - 1$).

$N_x = N_y$	$N_e(u)$	
	DF	IF
6	1.39×10^{0}	8.25×10^{-1}
8	3.48×10^{-1}	5.03×10^{-2}
10	6.78×10^{-2}	1.71×10^{-3}
12	6.92×10^{-3}	5.33×10^{-5}
14	4.78×10^{-4}	1.54×10^{-6}
16	2.40×10^{-5}	3.88×10^{-8}
18	9.19×10^{-7}	8.37×10^{-10}
20	2.76×10^{-8}	1.55×10^{-11}
22	6.87×10^{-10}	2.67×10^{-12}
24	2.96×10^{-10}	4.21×10^{-12}
	$O(N^{-13.02})$	$O(N^{-18.42})$

Results concerning the discrete relative L_2 error (N_e) of the solution u obtained by the integral and differential collocation formulations with Chebyshev polynomials are shown in Tables 1. The integral formulation outperforms the differential one regarding accuracy and convergence speed. For example, for the first eight sets, the integral and differential formulations converge as $O(N^{-18.42})$ and $O(N^{-13.02})$, respectively ($N = N_x - 1 = N_y - 1$).

Example 2 Consider the thermally-driven cavity flow in a square slot. For this problem, the governing equation presents the coupling of momentum (fourth-order PDE, streamfunction formulation) and energy (second-order PDE) equations. Very thin boundary layers are formed at high values of the Rayleigh number, thereby making the numerical simulation difficult. This problem provides a good means for testing and validating numerical methods. From the literature, a range of the Rayleigh number from 10^3 to 10^6 is usually employed for the verification of numerical methods. Table 2 and Figure 1 show that the present RBF method is able to produce highly accurate results for higher values of the Rayleigh number (e.g. $Ra = 10^7$).

Conclusion

In this paper, the integral collocation formulation is employed with 1D Chebyshev and RBF interpolation schemes for solving high-order PDEs. The convergence rates obtained are faster than those of the conventional differential formulation. In the case of RBFs, the computational costs of constructing the RBF approximations are significantly reduced when compared with the case of 2D integrated

Table 2: Example 2, RBFs, Natural convection, $Ra = 10^7$. A comparison of the present maximum horizontal and vertical velocities on the vertical and horizontal mid-planes of the cavity (u_{max} and v_{max}) and their locations, and the average Nusselt number throughout the cavity ($\overline{N_u}$) with the corresponding benchmark results [5].

Grid	u_{max}(error %)	y	v_{max}(error %)	x	$\overline{N_u}$(error %)
51×51	143.9(3.16)	0.885	680.8(2.63)	0.021	16.126(2.40)
61×61	146.9(1.14)	0.881	693.1(0.87)	0.021	16.370(0.93)
71×71	148.2(0.27)	0.879	698.0(0.17)	0.021	16.463(0.36)
81×81	148.8(0.13)	0.878	699.7(0.07)	0.021	16.499(0.15)
Benchmark [5]	148.6	0.879	699.2	0.021	16.523

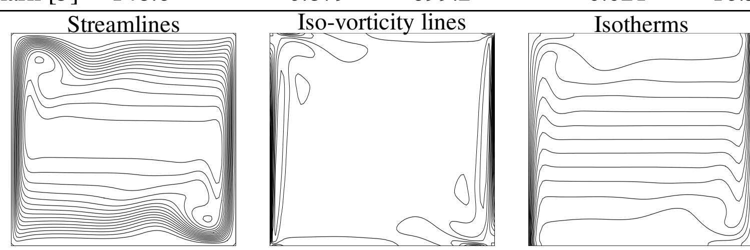

Figure 1: Natural convection: flow at $Ra = 10^7$ using 81×81.

RBF schemes, and hence larger numbers of nodes are able to be employed. Moreover, the condition numbers of the system matrix are also significantly improved. Finally, it is noted that the technique is also applicable for 3D problems which will be considered in future works.

Acknowledgement

This research is supported by the Australian Research Council.

References

1. Madych, W.R. and Nelson, S.A. (1988): "Multivariate interpolation and conditionally positive definite functions", *Approximation Theory and its Applications*, Vol. 4, pp. 77-89.

2. Trefethen, L.N. (2000): *Spectral Methods in MATLAB*, SIAM.

3. Mai-Duy, N. and Tran-Cong, T. (2001): "Numerical solution of differential equations using multiquadric radial basis function networks", *Neural Networks*, Vol. 14, pp. 185-199.

4. Mai-Duy, N. and Tran-Cong, T. (2005): "An efficient indirect RBFN-based method for numerical solution of PDEs", *Numerical Methods for Partial Differential Equations*, Vol 21, pp. 770-790.

5. Le Quere, P. (1991): "Accurate solutions to the square thermally driven cavity at high Rayleigh number", *Computers & Fluids*, Vol. 20(1), pp. 29-41.