

A study on convergence and complexity of reproducing kernel collocation method

Hsin-Yun Hu[†] and Chiu-Kai Lai

Department of Mathematics, Tunghai University, Taichung 407, Taiwan R.O.C.

Jiun-Shyan Chen

Department of Civil and Environmental Engineering, University of California, Los Angeles, CA 90095, U.S.A.

(Received June 1, 2009, Accepted August 14, 2009)

Abstract. In this work, we discuss a reproducing kernel collocation method (RKCM) for solving 2nd order PDE based on strong formulation, where the reproducing kernel shape functions with compact support are used as approximation functions. The method based on strong form collocation avoids the domain integration, and leads to well-conditioned discrete system of equations. We investigate the convergence and the computational complexity for this proposed method. An important result obtained from the analysis is that the degree of basis in the reproducing kernel approximation has to be greater than one for the method to converge. Some numerical experiments are provided to validate the error analysis. The complexity of RKCM is also analyzed, and the complexity comparison with the weak formulation using reproducing kernel approximation is presented.

Keywords: reproducing kernel approximation; convergence; complexity; strong form collocation method.

1. Introduction

The finite element method (FEM) (Ciarlet 1978) dominates the development of numerical methods for several decades due to its deep mathematical theory and wide applications. The meshfree methods (Belytschko *et al.* 1996) is a new class of computational method and they have received much attention in the last decade. Meshfree methods inherit most of the advantages of FEM, in the mean while, overcoming a major disadvantage in the discretization where no mesh is needed; the discretization in meshfree methods merely rely on a set of particle. Many meshfree methods were developed in recent years, such as smooth particle hydrodynamics (Gingold and Monaghan 1977), diffuse element method (DEM) (Nayroles *et al.* 1992), element free Galerkin (EFG) method (Belytschko *et al.* 1994), partition of unity method (PUM) (Melenk and Babuska 1996), HP-clouds (Duarte and Oden 1996), reproducing kernel particle method (RKPM) (Liu *et al.* 1995, Chen *et al.* 1996, Chen *et al.* 2000, Chen *et al.* 2003), natural element method (NEM)

[†] Assistant Professor, Corresponding Author, E-mail: huhy@thu.edu.tw

(Sukumar *et al.* 1998), meshless local Petrov-Galerkin method (MLPG) (Atluri and Zhu 2000), radial basis collocation method (RBCM) (Kansa 1992a, 1992b), multiple level set method (MLSM) (Zhang *et al.* 2008), among others. These methods are more effective than FEM in formulating adaptive refinement schemes, and in solving problems with cracks, large deformation, multi-scale features (Rojek and Onate 2008), and shocks.

The development of meshfree methods can be traced back from two branches, one based on weak Galerkin form, for example, DEM, EFG, PUM, RKPM, NEM, MLPG, MLSM, and the other based on strong collocation form, such as RBCM. In the weak form approach (Chen *et al.* 2003, Bernardi and Maday 1997, Han and Meng 2001) for 2nd order PDE, the approximation is in the Sobolev H^1 space, and it yields a symmetric discrete system when test and trial functions are constructed with the same approximation space. Using approximation functions with compact support, the resulting discrete system is stable like finite element method and it exhibits an algebraic convergence if the approximation functions have polynomial reproducibility (Chen *et al.* 1996, Chen *et al.* 2003). However, the need of quadrature rules in the domain integration consumes a higher computational cost. Alternatively, strong form collocation method (Hu and Li 2006, Li *et al.* 2008) avoids the need of domain integration, and the approximation is constructed in the Sobolev H^2 space. A commonly used approximation function for the strong form collocation method is the infinite differentiable radial basis function (Hardy 1971, 1990), and this method is generally called the radial basis collocation method that exhibits exponential convergence rate (Kansa 1992b, Hu and Li 2006). However the method is overshadowed by its fully dense and ill-conditioned discrete system due to the non-local nature of radial basis functions. In this work, we discuss a reproducing kernel collocation method (RKCM), where the reproducing kernel shape functions with compact support are used as approximation functions. The method avoids the domain integration, and leads to well-conditioned discrete system of equations.

The objective of this work is to study the convergence and complexity of a strong form collocation method based on a reproducing kernel approximation, called the reproducing kernel collocation method RKCM (Hu and Chen 2009). Reproducing kernel approximation functions (Liu *et al.* 1995, Chen *et al.* 1996, Han and Meng 2001) are compactly supported and RKCM yields a well-conditioned discrete system similar to that in the FEM, but it does not require a domain integration. We first discuss the convergence of this method and obtain the necessary conditions for convergence. We then analyze the complexity of the approach and make comparison with the condition in RKPM.

The remaining part of this paper is organized as follows. An introduction to reproducing kernel (RK) shape function, their properties and approximation error and convergence are given in Section 2. In Section 3, the derivatives of RK shape function, the formulation of RKCM and its convergence rate are discussed, and numerical experiments are illustrated to validate the theoretical results. The computational complexity is addressed in Section 4. Conclusions are summarized in the last Section.

2. Reproducing kernel (RK) approximation

2.1 The RK shape function

We first describe the RK approximation in one-dimension. The multi-dimensional formation can be easily obtained with similar construction. Let a function $f(x)$ can be approximated by

$$f^h(x) = \sum_{l=1}^{Np} d_l \psi_l(x), \quad x \in \Omega \quad (2.1)$$

where $\Omega \subset \mathbb{R}$, $\psi_l(x)$ are shape functions centered at x_l , and d_l are the coefficients to be sought. The construction of the shape functions is based on a set of particles

$$S = \{x_l\}_{l=1}^{Np} = \{x_1, x_2, \dots, x_{Np}\} \subseteq \Omega \cup \partial\Omega \quad (2.2)$$

with maximal nodal distance h , Np is the number of particles, and the RK shape function is given as follows:

$$\psi_l(x) = \mathbf{H}^T(0) \mathbf{M}^{-1}(x) \mathbf{H}(x - x_l) \phi_a(x - x_l) \quad (2.3)$$

where

$$\mathbf{M}(x) = \sum_{l=1}^{Np} \mathbf{H}(x - x_l) \mathbf{H}^T(x - x_l) \phi_a(x - x_l) \quad (2.4)$$

$$\mathbf{H}^T(x - x_l) = [1, x - x_l, (x - x_l)^2, \dots, (x - x_l)^n] \quad (2.5)$$

$$\mathbf{H}^T(0) = [1, 0, \dots, 0] \quad (2.6)$$

The vectors $\mathbf{H}(x - x_l)$ has dimension $n + 1$, and $\mathbf{M}(x)$ is a moment matrix with dimension $(n + 1) \times (n + 1)$. The function $\phi_a(x - x_l)$ is called the kernel function, for example, the cubic B-spline

$$\phi_a(z) = \begin{cases} \frac{2}{3} - 4z^2 + 4z^3, & 0 \leq z < \frac{1}{2} \\ \frac{4}{3} - 4z + 4z^2 - \frac{4}{3}z^3, & \frac{1}{2} \leq z < 1 \\ 0, & z \geq 1 \end{cases} \quad (2.7)$$

where $z = \frac{|x - x_l|}{a}$ and a is support size. For flexibility, we allow the support size to be dependent

on l . Another choice of kernel function is the quintic B-spline

$$\phi_a(z) = \begin{cases} \frac{11}{20} - \frac{9z^2}{2} + \frac{81z^4}{4} - \frac{81z^5}{4}, & 0 \leq z < \frac{1}{3} \\ \frac{17}{40} + \frac{15z}{8} - \frac{63z^2}{4} + \frac{135z^3}{4} - \frac{243z^4}{8} + \frac{81z^5}{8}, & \frac{1}{3} \leq z < \frac{2}{3} \\ \frac{81}{40} - \frac{81z}{8} + \frac{81z^2}{4} - \frac{81z^3}{4} + \frac{81z^4}{8} - \frac{81z^5}{40}, & \frac{2}{3} \leq z < 1 \\ 0, & z \geq 1 \end{cases} \quad (2.8)$$

Here we show how the RK shape function $\psi_l(x)$ in (2.3) is obtained. RK shape function can be

constructed as the multiplication of two functions

$$\psi_I(x) = C(x; x - x_I) \phi_a(x - x_I), \quad \forall x_I \in S \quad (2.9)$$

where $C(x; x - x_I)$ is a correction function and $\phi_a(x - x_I)$ is kernel function. For reproduction of complete n th order polynomial, the correction function is formed by a set of polynomial basis as

$$C(x; x - x_I) = \sum_{i=0}^n (x - x_I)^i b_i(x) = : \mathbf{H}^T(x - x_I) \mathbf{b}(x) \quad (2.10)$$

The coefficients $b_i(x)$ can be obtained by satisfying the following reproducing conditions

$$\sum_{I=1}^{Np} \psi_I(x) x_I^i = x^i, \quad i = 0, 1, \dots, n \quad (2.11)$$

that is,

$$\sum_{I=1}^{Np} C(x; x - x_I) \phi_a(x - x_I) x_I^i = x^i, \quad i = 0, 1, \dots, n \quad (2.12)$$

where n is reproducing degree. Eq. (2.12) is equivalent to

$$\sum_{I=1}^{Np} C(x; x - x_I) \phi_a(x - x_I) (x - x_I)^i = \delta_{i0}, \quad i = 0, 1, \dots, n \quad (2.13)$$

where δ_{ij} is Kronecker delta. Eq. (2.13) can be rewritten in a vector form

$$\sum_{I=1}^{Np} C(x; x - x_I) \phi_a(x - x_I) \mathbf{H}(x - x_I) = \mathbf{H}(0) \quad (2.14)$$

Substituting (2.10) into (2.14), we obtain

$$\sum_{I=1}^{Np} \mathbf{H}(x - x_I) \mathbf{H}^T(x - x_I) \phi_a(x - x_I) \mathbf{b}(x) = \mathbf{H}(0) \quad (2.15)$$

Denote above equation as

$$\mathbf{M}(x) \mathbf{b}(x) = \mathbf{H}(0) \quad (2.16)$$

where $\mathbf{M}(x)$ is defined as

$$\mathbf{M}(x) = \begin{bmatrix} \sum_{I=1}^{Np} \phi_a(x - x_I) & \sum_{I=1}^{Np} (x - x_I) \phi_a(x - x_I) & \dots & \sum_{I=1}^{Np} (x - x_I)^n \phi_a(x - x_I) \\ \vdots & \vdots & \vdots & \vdots \\ \sum_{I=1}^{Np} (x - x_I)^n \phi_a(x - x_I) & \sum_{I=1}^{Np} (x - x_I)^{n+1} \phi_a(x - x_I) & \dots & \sum_{I=1}^{Np} (x - x_I)^{2n} \phi_a(x - x_I) \end{bmatrix} \quad (2.17)$$

with dimension $(n+1) \times (n+1)$. From (2.16), we have the coefficient vector

$$\mathbf{b}(x) = \mathbf{M}^{-1}(x) \mathbf{H}(0) \quad (2.18)$$

The correction function in (2.10) becomes

$$C(x; x - x_I) = \mathbf{H}^T(x - x_I) \mathbf{M}^{-1}(x) \mathbf{H}(0) = \mathbf{H}^T(0) \mathbf{M}^{-1}(x) \mathbf{H}(x - x_I) \quad (2.19)$$

Thus, we have the function approximation

$$\begin{aligned} f^h(x) &= \sum_{I=1}^{Np} d_I \psi_I(x) = \sum_{I=1}^{Np} d_I C(x; x - x_I) \phi_a(x - x_I) \\ &= \sum_{I=1}^{Np} d_I \mathbf{H}^T(0) \mathbf{M}^{-1}(x) \mathbf{H}(x - x_I) \phi_a(x - x_I) \end{aligned} \quad (2.20)$$

where

$$\psi_I(x) = \mathbf{H}^T(0) \mathbf{M}^{-1}(x) \mathbf{H}(x - x_I) \phi_a(x - x_I) \quad (2.21)$$

This shape function $\psi_I(x)$ is called the *reproducing kernel (RK) shape function*, and its properties are summarized as follows.

- (a) The construction of RK shape function $\psi_I(x)$ does not rely on a mesh, it only relies a set of particles.
- (b) The RK shape function has the same compact support as the kernel function, and the smoothness and locality of the shape function is determined by the kernel function.
- (c) The degree of completeness and consistency of the shape function is determined by the correction function.
- (d) For the moment matrix $\mathbf{M}(x)$ to be nonsingular, any position $x \in \Omega$ needs to be covered by at least $n + 1$ kernel functions. Let κ be the maximal overlapping number in domain, usually, $\kappa \geq n + 1$. A suggestion for the choice of support size is $a = (n + 1)h$, where h is maximal nodal distance.
- (e) The RK shape function does not satisfy the Kronecker delta property, i.e., $\psi_I(x_J) \neq \delta_{IJ}$. This means that the coefficients are not equal to nodal values, i.e., $f^h(x_I) \neq d_I$.

2.2 Convergence properties of RK approximation

In this section we study the convergence of function approximation by RK shape functions.

Assume $f(x) = \sin \pi x$, where $x \in [0, 1] =: \Omega$. The sine function can be approximated by

$$f^h(x) = \sum_{I=1}^{Np} d_I \psi_I(x), \quad x \in \Omega \quad (2.22)$$

in which the cubic B-spline kernel function $\phi_a(x - x_I)$ is adopted in the RK shape function. The set of nodal points $\{x_I\}_{I=1}^{Np}$ is called the source points. We define another set of points, called the collocation points

$$\Xi = \{\xi_i\}_{i=1}^{Ns} = \{\xi_1, \xi_2, \dots, \xi_{Ns}\} \subseteq \Omega \cup \partial\Omega \quad (2.23)$$

where Ns is the total number of collocation points. The set of collocation points may or may not equal to the set of source points used in the approximation in (2.2). To approximate the function $f(x)$, we force the residuals of the function approximation to be zero at these collocation points:

$$f^h(\xi_i) - f(\xi_i) = 0, \quad \forall \xi_i, \quad i = 1, 2, \dots, Ns \quad (2.24)$$

It follows that

$$f^h(\xi_i) = \sin \pi \xi_i, \forall \xi_i \quad (2.25)$$

And this can be written a linear system as

$$f^h(\xi_1) = \sum_{I=1}^{Np} d_I \psi_I(\xi_1) = \sin \pi \xi_1 \quad (2.26)$$

$$f^h(\xi_2) = \sum_{I=1}^{Np} d_I \psi_I(\xi_2) = \sin \pi \xi_2 \quad (2.27)$$

$$\vdots$$

$$f^h(\xi_{Ns}) = \sum_{I=1}^{Np} d_I \psi_I(\xi_{Ns}) = \sin \pi \xi_{Ns} \quad (2.28)$$

Denote above system of equations as follows

$$\begin{bmatrix} \psi_1(\xi_1) & \psi_2(\xi_1) & \cdots & \psi_{Np}(\xi_1) \\ \psi_1(\xi_2) & \psi_2(\xi_2) & \cdots & \psi_{Np}(\xi_2) \\ \vdots & \vdots & \vdots & \vdots \\ \psi_1(\xi_{Ns}) & \psi_2(\xi_{Ns}) & \cdots & \psi_{Np}(\xi_{Ns}) \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_{Np} \end{bmatrix} = \begin{bmatrix} \sin \pi \xi_1 \\ \sin \pi \xi_2 \\ \vdots \\ \sin \pi \xi_{Ns} \end{bmatrix} \quad (2.29)$$

When $Ns = Np = N$, the coefficients d_I can be obtain by Gaussian Elimination with Backward Substitution (GE with BS), and then we obtain the approximation $f^h(x)$. Since the RK approximation belongs to a local approximation, the support size of the RK shape function is selected as $a = (n+1)h$, this leads to a band matrix on the left hand side of linear system (2.29). A band Gaussian Elimination with Backward Substitution can be introduced to solve the linear system. The detailed algorithms are given in Appendix.

Here is a comparison of the operation counts for two Elimination methods mentioned above. For traditional Gaussian Elimination with Backward Substitution:

$$\text{M/D: } \frac{N^3}{3} + N^2 - \frac{N}{3} \quad (2.30)$$

$$\text{A/S: } \frac{N^3}{3} + \frac{N^2}{2} - \frac{5N}{6} \quad (2.31)$$

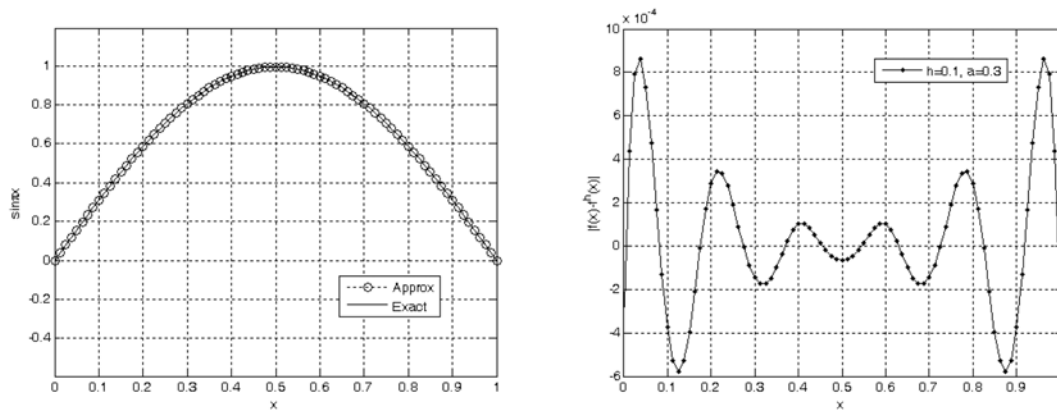
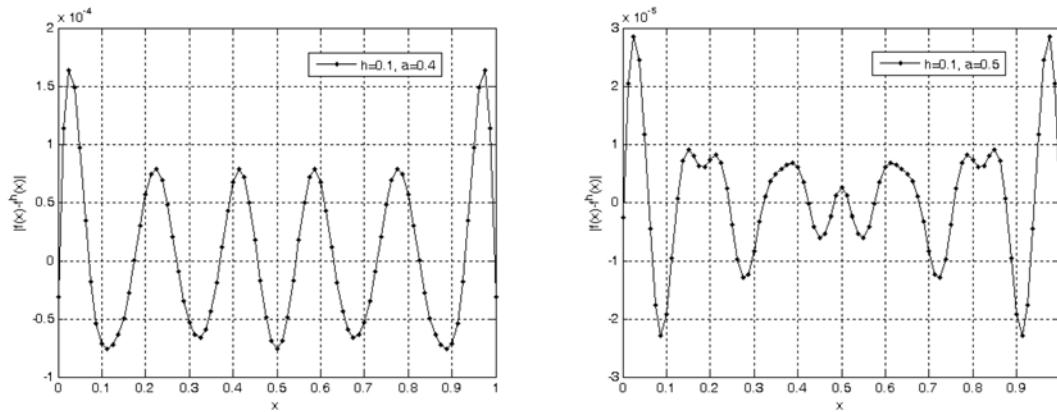
where the abbreviations M/D and A/S stand for multiplication/division and addition/ subtraction, respectively. For band Gaussian Elimination with Backward Substitution:

$$\text{M/D: } N(w^2 + w - 1) - \frac{2w^3}{3} + \frac{2w}{3} \quad (2.32)$$

$$\text{A/S: } N(w^2 - 1) - \frac{2w^3}{3} + \frac{w^2}{2} + \frac{w}{6} \quad (2.33)$$

Table 1 A comparison of the counts for two GE with BS

N	traditional GE with BS		w	band GE with BS	
	M/D	A/S		M/D	A/S
10	430	375	3	94	63
50	44,150	42,875	5	1,370	1,130
100	343,300	338,250	7	5,276	4,597


Fig. 1 The approximated sine function and the error by using support size $a = 0.3$ and degree of basis functions $n = 2$

Fig. 2 Errors of the approximated function obtained by using support sizes $a = 0.4$ and $a = 0.5$, and degree of basis functions $n = 2$

where the bandwidth of the band matrix is $2w - 1$. The operation counts under different dimensions $N = 10, 50, 100$ are listed in Table 1. Usually $w \ll N$, and the latter algorithm will be used for solving the linear system arising from RK approximation.

When $Ns > Np$, the coefficients d_l can be obtained by QR factorization based on the Householder and Givens transformation or by singular value decomposition (SVD) with substitutions (Golub and Van Loan 1996).

In the followings we provide numerical tests for the RK approximation, wherein equally spaced source points and collocation points are chosen

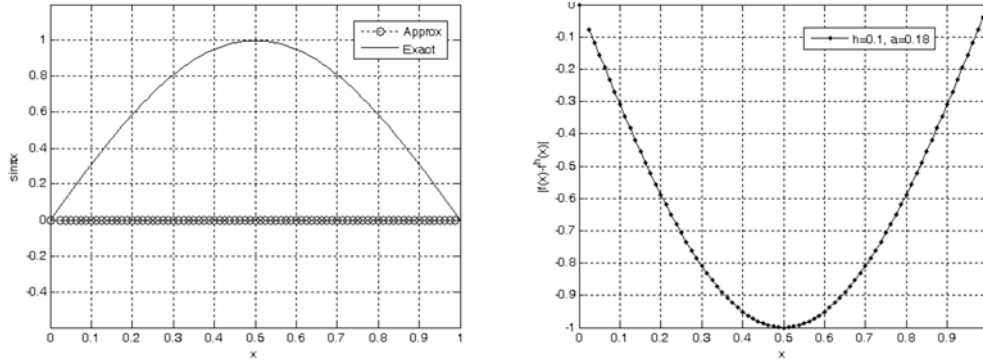


Fig. 3 Error of the approximated function obtained by using support size $a = 0.18$ and degree of basis functions $n = 2$

$$S = \{x_I = (I-1)/(Np-1)\}_{I=1}^{Np}, Np = 11 \quad (2.34)$$

and

$$\Xi = \{\xi_i = (i-1)/(Ns-1)\}_{i=1}^{Ns}, Ns = 2 \times Np - 1 \quad (2.35)$$

The number of collocation points is approximately two times of the number of source points. The degree of basis function $n = 2$ is considered with different support sizes. The exact function, the approximated functions and errors are plotted in Figs. 1 and 2.

If we consider much smaller support size for the quadratic basis function, we obtain a different result, see Fig. 3.

We can see from the result in Fig. 3 that the support size of kernel function has to be greater than the maximal nodal distance to obtain a solution, otherwise, it leads to a singular moment matrix and yields a wrong solution. The numerical results show that the larger support sizes does not seem to improve much on accuracy, but it significantly increases the computational cost due to the larger bandwidth in the system of Eq. (2.29).

The main parameters for accuracy are the number of source points Np and the degree of basis function n . We will discuss these issues in next section.

2.3 Convergence of RK approximation

We assume a quasi-uniform distribution of source points as defined below

$$c_0 \leq \frac{a_K}{a_J} \leq c_1, \forall K, J \quad (2.36)$$

where a_K and a_J denote the support sizes of kernel functions centered at source points x_K and x_J , respectively, and c_0 and c_1 are generic constants. If function $f(x)$ is $n+1$ order differentiable, then we have a local interpolation error (Chen *et al.* 2003, Han and Meng 2001) as follows

$$\|f(x) - f^h(x)\|_{0, \omega_I} \leq Ca_I^{n+1} |f(x)|_{n+1, \omega_I} \quad (2.37)$$

where ω_I is the compact support of kernel function centered at x_I , with support size a_I . The norm $\|\cdot\|_{0, \omega_I}$ denotes the Sobolev zero norm, which is equal to the L_2 -norm, and n is the degree of basis function.

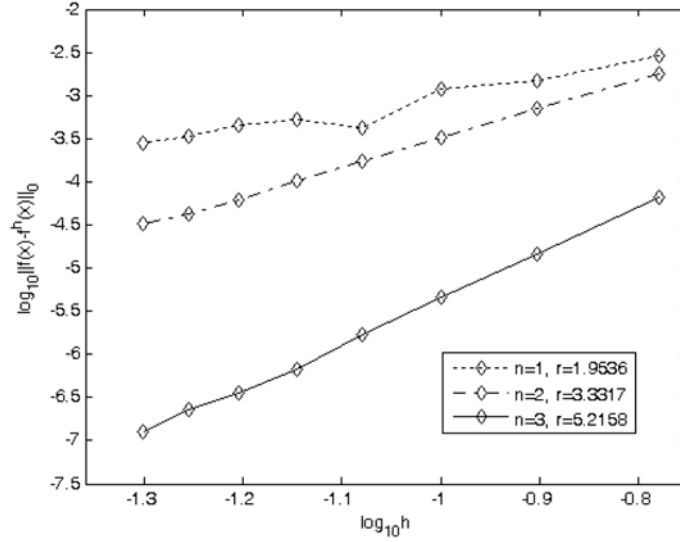


Fig. 4 Convergence behavior of RK approximation with different degrees of basis

Moreover, we have the following global interpolation error estimate.

Lemma 1. Let source points be in a quasi-uniform distribution. From the local interpolation error given in Eq. (2.37), we have the global interpolation error estimation

$$\|f(x) - f^h(x)\|_{0,\Omega} \leq C \kappa a^{n+1} |f(x)|_{n+1,\Omega} \quad (2.38)$$

where $\kappa \ll Np$ is the maximal overlapping number in domain, and $a = \max_I a_I$, and C is a constant independent of parameters κ , a and n .

Remark 1. If we choose $a = (n+1)h$ as the support size of kernel function, then we have the estimation

$$\|f(x) - f^h(x)\|_{0,\Omega} = O(a^{n+1}) \approx O(h^{n+1}) \quad (2.39)$$

where $n \geq 1$. The convergence rate is proportional to power $n+1$ of nodal distance h .

To validate the convergence rate, we use the reproducing degree $n = 1, 2, 3$, the number of source points $Np = 6, 8, \dots, 20$ and the number of collocation points $Ns = 12, 16, \dots, 40$. Equally spaced collocation points and source point are adopted. We define $h = 1/Np$ and choose $a = (n+1)h$. The result of errors in L_2 -norm with various levels of refinement is plotted in Fig. 4. Here r denotes the numerical order of convergence rate. Compared with the error estimate in Eq. (2.39), there exists a super-convergence behavior when the degree of basis function $n = 3$.

3. Reproducing kernel collocation method (RKCM)

In this section, the RK approximation described in Section 2 is used for solving partial differential equations under a strong form collocation framework, and we call it the *reproducing kernel collocation method (RKCM)*. The higher order derivatives of RK shape functions are first introduced, and the implementation scheme of RKCM and convergence properties are discussed.

3.1 Derivatives of RK shape function

The RK shape function is constructed by the multiplication of correction and kernel functions as

$$\psi_I(x) = C(x; x - x_I) \phi_a(x - x_I) \quad (3.1)$$

The first and second order derivatives can be obtain by product rule

$$\psi_{I,x}(x) = C_{,x}(x; x - x_I) \phi_a(x - x_I) + C(x; x - x_I) \phi_{a,x}(x - x_I) \quad (3.2)$$

$$\begin{aligned} \psi_{I,xx}(x) = & C_{,xx}(x; x - x_I) \phi_a(x - x_I) + 2C_{,x}(x; x - x_I) \phi_{a,x}(x - x_I) \\ & + C(x; x - x_I) \phi_{a,xx}(x - x_I) \end{aligned} \quad (3.3)$$

where the correction function and its derivatives are given as

$$C(x; x - x_I) = \mathbf{H}^T(0) \mathbf{M}^{-1}(x) \mathbf{H}(x - x_I) \quad (3.4)$$

$$C_{,x}(x; x - x_I) = \mathbf{H}^T(0) \mathbf{M}_{,x}^{-1}(x) \mathbf{H}(x - x_I) + \mathbf{H}^T(0) \mathbf{M}^{-1}(x) \mathbf{H}_{,x}(x - x_I) \quad (3.5)$$

$$\begin{aligned} C_{,xx}(x; x - x_I) = & \mathbf{H}^T(0) \mathbf{M}_{,xx}^{-1}(x) \mathbf{H}(x - x_I) + 2\mathbf{H}^T(0) \mathbf{M}_{,x}^{-1}(x) \mathbf{H}_{,x}(x - x_I) \\ & + \mathbf{H}^T(0) \mathbf{M}^{-1}(x) \mathbf{H}_{,xx}(x - x_I) \end{aligned} \quad (3.6)$$

The moment matrix and its derivatives are given as

$$\mathbf{M}(x) = \sum_{I=1}^{NP} \mathbf{H}(x - x_I) \mathbf{H}^T(x - x_I) \phi_a(x - x_I) \quad (3.7)$$

$$\begin{aligned} \mathbf{M}_{,x}(x) = & \sum_{I=1}^{NP} \{ \mathbf{H}_{,x}(x - x_I) \mathbf{H}^T(x - x_I) \phi_a(x - x_I) + \mathbf{H}(x - x_I) \mathbf{H}_{,x}^T(x - x_I) \phi_a(x - x_I) \\ & + \mathbf{H}(x - x_I) \mathbf{H}^T(x - x_I) \phi_{a,x}(x - x_I) \} \end{aligned} \quad (3.8)$$

$$\begin{aligned} \mathbf{M}_{,xx}(x) = & \sum_{I=1}^{NP} \{ \mathbf{H}_{,xx}(x - x_I) \mathbf{H}^T(x - x_I) \phi_a(x - x_I) + 2\mathbf{H}_{,x}(x - x_I) \mathbf{H}_{,x}^T(x - x_I) \phi_a(x - x_I) \\ & + 2\mathbf{H}_{,x}(x - x_I) \mathbf{H}^T(x - x_I) \phi_{a,x}(x - x_I) + \mathbf{H}(x - x_I) \mathbf{H}_{,xx}^T(x - x_I) \phi_a(x - x_I) \\ & + 2\mathbf{H}(x - x_I) \mathbf{H}_{,x}^T(x - x_I) \phi_{a,x}(x - x_I) + \mathbf{H}(x - x_I) \mathbf{H}^T(x - x_I) \phi_{a,xx}(x - x_I) \} \end{aligned} \quad (3.9)$$

In the followings, we use the implicit differentiation to obtain the derivatives of $\mathbf{M}^{-1}(x)$, i.e., $\mathbf{M}_{,x}^{-1}(x)$ and $\mathbf{M}_{,xx}^{-1}(x)$. Since $\mathbf{M}(x) \mathbf{M}^{-1}(x) = \mathbf{I}$, where \mathbf{I} denotes an identify matrix. Taking differentiation on both sides, we obtain

$$\frac{d}{dx}(\mathbf{M}(x) \mathbf{M}^{-1}(x)) = \frac{d}{dx} \mathbf{I} \quad (3.10)$$

and then by product rule

$$\mathbf{M}_{,x}(x) \mathbf{M}^{-1}(x) + \mathbf{M}(x) \mathbf{M}_{,x}^{-1} = \mathbf{0} \quad (3.11)$$

where $\mathbf{0}$ is a zero matrix, it follows that

$$\mathbf{M}_{,x}^{-1} = -\mathbf{M}^{-1}(x)\mathbf{M}_{,x}(x)\mathbf{M}^{-1}(x) \quad (3.12)$$

In a similar manner, we may obtain the second derivative

$$\frac{d}{dx}(\mathbf{M}_{,x}^{-1}) = \frac{d}{dx}(-\mathbf{M}^{-1}(x)\mathbf{M}_{,x}(x)\mathbf{M}^{-1}(x)) \quad (3.13)$$

and then

$$\mathbf{M}_{,xx}^{-1} = -\mathbf{M}^{-1}(x)(\mathbf{M}_{,xx}(x)\mathbf{M}^{-1}(x) + 2\mathbf{M}_{,x}(x)\mathbf{M}_{,x}^{-1}(x)) \quad (3.14)$$

In section 4, we will discuss the complexity of the shape function and its derivatives based on the equations given in Eq. (3.1) to Eq. (3.14). For the construction of the shape function and its derivatives, we first form the matrices $\mathbf{M}^{-1}(x)$, $\mathbf{M}_{,x}^{-1}(x)$, $\mathbf{M}_{,xx}^{-1}(x)$, then the correction function and derivatives $C(x; x - x_I)$, $C_{,x}(x; x - x_I)$, $C_{,xx}(x; x - x_I)$ and finally the shape function and derivatives $\psi_I(x)$, $\psi_{I,x}(x)$, $\psi_{I,xx}(x)$ can be constructed.

Denote $\psi_I^{(1)}(x) = \psi_{I,x}(x)$ and $\psi_I^{(2)}(x) = \psi_{I,xx}(x)$, we have the bounds (Chen *et al.* 2003, Han and Meng 2001)

$$\|\psi_I(x)\|_{\infty} < C_1 \quad (3.15)$$

$$\|\psi_I^{(l)}(x)\|_{\infty} \leq C_2 a^{-l}, \quad l = 1, 2, \dots, \quad (3.16)$$

where C_1 and C_2 are generic constants. In the followings we will provide the bounds for the high order derivatives of a function $v = v(x)$, which is approximated by the linear combination of RK shape function

$$v(x) = \sum_{I=1}^{N_p} b_I \psi_I(x) \quad (3.17)$$

Denote finite dimensional space

$$V = \text{span}\{\psi_1(x), \psi_2(x), \dots, \psi_{N_p}(x)\} \subset H^2(\Omega) \quad (3.18)$$

We need the following inverse inequalities to prove the convergence.

Lemma 2. Let the source points be in a quasi-uniform distribution. For $\forall v \in V$, there exist the inequalities

$$\|v\|_{l,\Omega} \leq C_1 \kappa^{1/2} a^{-l} n^{2l} \|v\|_{0,\Omega}, \quad l = 1, 2, 3, \dots, \quad (3.19)$$

$$\|v\|_{l,\Gamma} \leq C_2 \kappa^{1/2} a^{-l} n^{2l} \|v\|_{1,\Omega}, \quad l = 1, 2, 3, \dots, \quad (3.20)$$

$$\|v_{,x}\|_{l,\Gamma} \leq C_3 \kappa^{1/2} a^{-(l+1)} n^{2(l+1)} \|v\|_{1,\Omega}, \quad l = 1, 2, 3, \dots, \quad (3.21)$$

where κ is the maximal overlapping number in domain, a is the maximal support size, $\Gamma = \partial\Omega$ denotes the boundary of Ω , and C_i are generic constants.

3.2 Implementation scheme for RKCM

Consider an elliptic problem

$$L_d u = f, \text{ in } \Omega \quad (3.22)$$

$$B_d u = g, \text{ on } \Gamma \quad (3.23)$$

where L_d and B_d denote the operators in the domain and on the boundary, respectively.

The collocation method is the minimization of discrete least-squares functional (Hu and Li 2006, Li *et al.* 2008)

$$\hat{E}(u_k) = \min_{v \in V} \hat{E}(v) \quad (3.24)$$

where $\hat{E}(v)$ is a quadrature version of least-squares functional defined as

$$\hat{E}(v) = \frac{1}{2} \hat{\int}_{\Omega} (L_d v - f)^2 d\Omega + \frac{1}{2} \hat{\int}_{\Gamma} (B_d v - g)^2 d\Gamma \quad (3.25)$$

here $\hat{\int}$ denotes the quadrature version of \int . A set of collocation points $\Xi = \{\xi_1, \xi_2, \dots, \xi_{N_s}\}$ with maximal spacing $\bar{h} = \max_{\xi_i \in \bar{\Omega}} \{\xi_i\}$ is used. The parameter \bar{h} has to satisfy the following relationship (Hu and Li 2006)

$$\bar{h} \simeq o(a^2) \approx o(h^2) \quad (3.26)$$

to ensure the error from quadrature rules to be bounded as follows

$$\left| \left(\int - \hat{\int} \right) G(x) \right| = o(1) \rightarrow 0 \quad (3.27)$$

Here the integrand $G(x)$ can be $L_d v(x)$ or $B_d v(x)$. Consequently we may obtain an optimal solution $u_k(x)$ as discussed below.

The minimization of discrete functional (3.24) leads to solving the following system of equations:

$$L_d \left\{ \sum_{I=1}^{N_p} b_I \psi_I(\xi_i) \right\} = f(\xi_i), \quad \xi_i \in \Omega \quad (3.28)$$

$$B_d \left\{ \sum_{I=1}^{N_p} b_I \psi_I(\xi_j) \right\} = g(\xi_j), \quad \xi_j \in \Gamma \quad (3.29)$$

They can be rewritten as follows

$$\mathbf{F}\mathbf{x} = \begin{bmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_{N_p} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \end{bmatrix} =: \mathbf{r} \quad (3.30)$$

The entries of matrix \mathbf{F} and the components of \mathbf{r} are given by

$$[\mathbf{F}_1]_{Ii} = \mathbf{L}_d \psi_I(\xi_i), [\mathbf{r}_1]_i = f(\xi_i), i = 1, 2, \dots, Nt \quad (3.31)$$

$$[\mathbf{F}_2]_{Ij} = \mathbf{B}_d \psi_I(\xi_j), [\mathbf{r}_2]_j = g(\xi_j), j = Nt + 1, \dots, Ns \quad (3.32)$$

where $I = 1, 2, \dots, Np$, Nt is the number of collocation points in the domain, and Ns is the total number of collocation points in the domain and on the boundaries. The matrix \mathbf{F} has dimension $Ns \times Np$, \mathbf{r} is a known vector with dimension Ns . We have an over-determined system when $Ns > Np$, and the Cholesky decomposition with substitution (Burden and Faires 2005) can be adopted on its normal equation to find coefficients b_p and to obtain an optimal solution $u_k(x) \in V$.

3.3 Convergence of RBCM

In this section let us discuss the convergence properties of RKCM. Consider the following linear operators in one-dimension

$$\mathbf{L}_d = \frac{d^2}{dx^2} \quad (3.33)$$

$$\mathbf{B}_d = \alpha \frac{d}{dx} + \beta \quad (3.34)$$

where $\alpha \geq 0, \beta > 0$. Define a norm

$$\|v\|_H = \{\|v\|_{2,\Omega}^2 + \|\alpha v_x + \beta\|_{0,\Gamma}^2\}^{\frac{1}{2}} \quad (3.35)$$

The minimization of functional Eq. (3.24) can be described equivalently in the following variational problem

$$\hat{B}(u, v) = \hat{F}(v) \quad (3.36)$$

where

$$\hat{B}(u, v) = \int_{\Omega} u_{,xx} v_{,xx} d\Omega + \int_{\Gamma} (\alpha u_{,x} + \beta)(\alpha v_{,x} + \beta) dl \quad (3.37)$$

$$\hat{F}(v) = -\int_{\Omega} f v_{,xx} d\Omega + \int_{\Gamma} g(\alpha v_{,x} + \beta) dl \quad (3.38)$$

Lemma 3. Suppose there exist two inequalities (Hu and Chen 2009),

$$\hat{B}(u, v) \leq C_1 \|u\|_H \|v\|_H, \forall v \in V \quad (3.39)$$

$$\hat{B}(v, v) \geq C_2 \|v\|_H^2, \forall v \in V \quad (3.40)$$

Then, there exists an optimal estimate

$$\|u - u_k\|_H \leq C_3 \min_{v \in V} \|u - v\|_H \quad (3.41)$$

where C_i are constants independent of a, h, n and Np .

Lemma 4. Assume that the solution $u(x)$ is sufficiently smooth, the global interpolation error estimate for solution of RKCM is given as follows

$$\|u - u_k\|_{l, \Omega} \leq C \kappa a^{n+1-l} |u|_{n+1, \Omega} \quad (3.42)$$

where $l \geq 1$, and C is a constant independent of κ , a and n .

Theorem 1. Let the conditions in Lemmas 2, 3 and 4 hold. There exists the error bound for the solution of RKCM as follows (Hu and Chen 2009)

$$\begin{aligned} \|u - u_k\|_H &\leq C \{ \|u - v\|_{2, \Omega} + \alpha \|u - v\|_{1, \Gamma} + \beta \|u - v\|_{0, \Gamma} \} \\ &\leq C \kappa a^{n-1} |u|_{n+1, \Omega} \end{aligned} \quad (3.43)$$

where κ is the overlapping number, a is the maximal support size and n is the degree of basis function, and C is a constant.

According to Theorem 1, there exists the following convergence behavior

$$\|u - u_k\|_H = O(1), \text{ when } n = 1 \quad (3.44)$$

This shows that the error does not converge when the linear basis function $n=1$ is adopted. Specifically, the degree of basis function has to be greater than one, $n > 1$, for convergence. Moreover, we have

$$\|u - u_k\|_H = O(a^1) \approx O(h^1), \text{ when } n = 2 \quad (3.45)$$

$$\|u - u_k\|_H = O(a^2) \approx O(h^2), \text{ when } n = 3 \quad (3.46)$$

The error of solution under H -norm is equivalent to in Sobolev two norm, it follows that the convergence behavior in Sobolev zero norm will be

$$\|u - u_k\|_{0, \Omega} = O(a^{1+2}) \approx O(h^3), \text{ when } n = 2 \quad (3.47)$$

$$\|u - u_k\|_{0, \Omega} = O(a^{2+2}) \approx O(h^4), \text{ when } n = 3 \quad (3.48)$$

Two numerical tests will be given in the next section to validate these theoretical results.

3.4 Numerical experiments

Consider a Poisson problem in domain $\Omega = \{x | 0 < x < 1\}$ with pure Dirichlet boundary conditions

$$\begin{aligned} u''(x) &= f(x), \text{ in } \Omega \\ u(0) &= 0, \text{ on } \Gamma_0 \\ u(1) &= 0, \text{ on } \Gamma_1 \end{aligned} \quad (3.49)$$

where $u''(x) = d^2 u(x) / dx^2$, $f(x) = -\pi^2 \sin \pi x$, Γ_0 denotes endpoint $x=0$, and Γ_1 denotes endpoint $x=1$. The analytical solution is $u(x) = \sin \pi x$.

We use the reproducing degree $n = 1, 2, 3$, the number of source points $Np = 6, 8, \dots, 20$, and

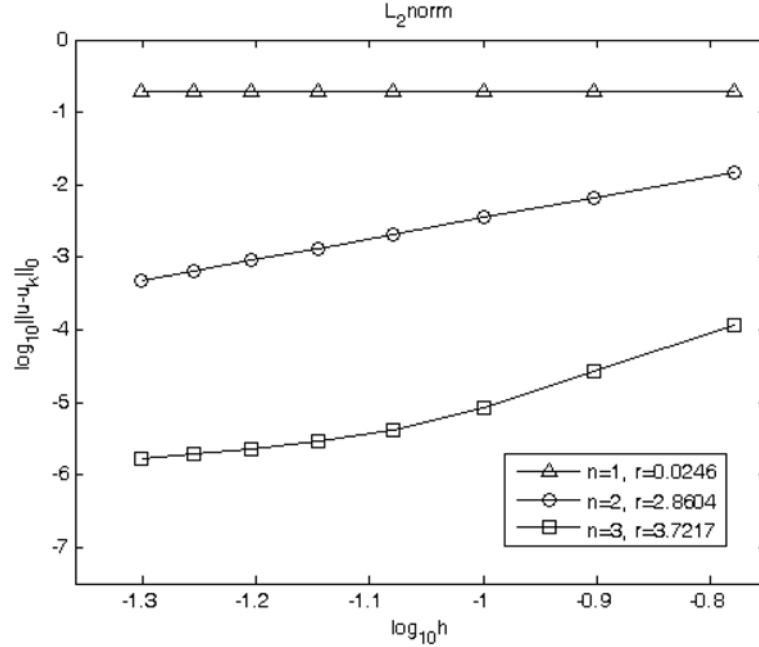


Fig. 5 Convergence behavior for Poisson's problem with pure Dirichlet boundary conditions

the number of collocations is four times of the number of source points, i.e., $N_s = 24, 32, \dots, 80$. Equally spaced collocation points and source points are used, and the kernel function $\phi_a(x - x_i)$ is chosen as the quintic B-spline defined in Eq. (2.8). We define $h = 1/N_p$ and choose the support size $a = (n + 1)h$. The errors of solution in L_2 - norm with various levels of refinement are plotted in Fig. 5.

In the second test, we consider a problem with different boundary conditions as follows

$$\begin{aligned} u''(x) &= f(x), \text{ in } \Omega \\ u(0) &= 0, \text{ on } \Gamma_0 \\ u'(1) &= -\pi, \text{ on } \Gamma_1 \end{aligned} \quad (3.50)$$

where $u'(x) = du(x)/dx$. The exact solution is the same as previous example. The same reproducing degree, collocation points and source points as those used in the previous example are employed in this experiment. The test results are illustrated in Fig. 6.

We can see from Figs. 5 and 6 that the error, indeed, does not converge when linear basis function $n = 1$ is used. This is consistent with our theoretical results. The rates of convergence for $n = 2, 3$ are reasonably consistent with the theoretical results in Eqs. (3.47) and (3.48). This convergence behavior is different from the that in the reproducing kernel particle method (RKPM) (Liu *et al.* 1995, Chen *et al.* 1996), in which the linear basis function is sufficient for convergence. Compared to weak formulation such as RKPM, RKCM based on strong form approach is rather simple in forming the discrete equation and it avoids domain integration. However, the second order derivatives are needed in RKCM. In the next section, we analyze the computational complexity of RKCM.

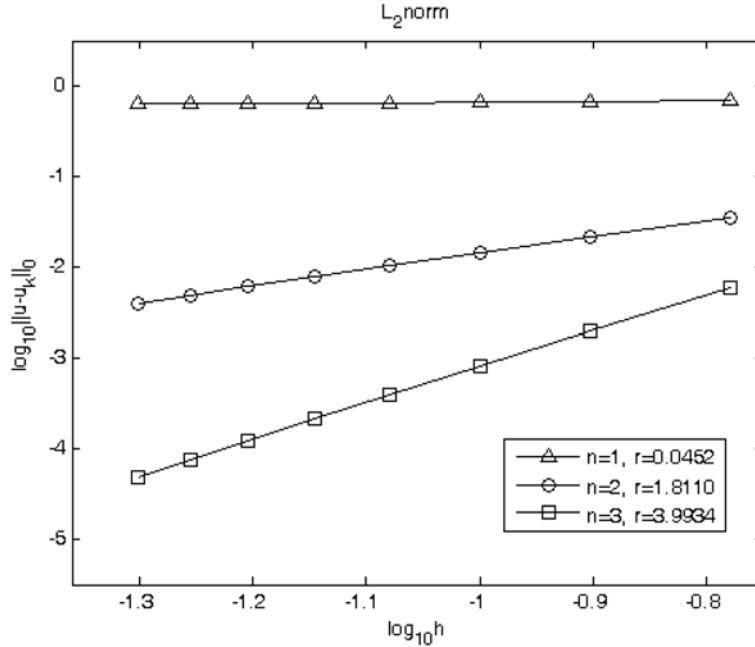


Fig. 6 Convergence behavior for Poisson's problem with Dirichlet and Neumann boundary conditions

4. Computational complexity of RKCM

In computations, the CPU time is proportional to the total operations in addition (A), subtraction (S), multiplication (M) and division (D) computations. In this section we discuss the operations counts for forming a system of Eq. (3.30) in RKCM and solution of the linear system.

The linear system Eq. (3.30) consists of an $Ns \times Np$ matrix \mathbf{F} and a vector \mathbf{r} with dimension Ns . Note that the main entries in matrix \mathbf{F} are $\psi_{I,xx}(\xi_i)$ in the domain, and $\psi_I(\xi_i)$ or $\psi_{I,x}(\xi_i)$ on the boundaries. First, we calculate the operation counts for forming RK shape function $\psi_I(\xi_i)$ and its derivatives $\psi_{I,x}(\xi_i)$ and $\psi_{I,xx}(\xi_i)$. The derivation was given in Section 3.1, the formation of RK shape function involves considerable vector-vector and vector-matrix multiplications. We will start with the operation count for the elementary operations and then the count for the construction of RK shape functions.

4.1 Elementary operations

Let \mathbb{R} denote the set of real values, \mathbb{R}^s denote the set of real vectors, and $\mathbb{R}^{s \times s}$ denote the set of $s \times s$ matrices.

(a) Dot product:

If $\mathbf{x}, \mathbf{y} \in \mathbb{R}^s$, where $\mathbf{x} = (x_1, x_2, \dots, x_s)^T$, $\mathbf{y} = (y_1, y_2, \dots, y_s)^T$ operation counts of the dot product,

$$\mathbf{x} \bullet \mathbf{y} = \mathbf{x}^T \mathbf{y} = \sum_{i=1}^s x_i y_i, \text{ are given below}$$

	M/D	A/S
Forming $\mathbf{x} \bullet \mathbf{y}$	s	$s - 1$

(b) Scale-Matrix multiplication:

If $\mathbf{A} \in \mathbb{R}^{s \times s}$, $\gamma \in \mathbb{R}$ where $\mathbf{A} = [a_{ij}]$, then operation count for the scale-matrix multiplication, $\gamma \cdot \mathbf{A} = \gamma \cdot [a_{ij}]$, $i, j = 1, 2, \dots, s$, is given as follows

	M/D	A/S
Forming $\gamma \cdot \mathbf{A}$	s^2	0

(c) Matrix-Vector multiplication:

If $\mathbf{A} \in \mathbb{R}^{s \times s}$, $\mathbf{x} \in \mathbb{R}^s$ where $\mathbf{A} = [a_{ij}]$ and $\mathbf{x} = (x_1, x_2, \dots, x_s)^T$, then each component of $\mathbf{A} \mathbf{x}$, i.e., $(\mathbf{A} \mathbf{x})_i = \sum_{j=1}^s a_{ij} x_j$, $i = 1, 2, \dots, s$, can be viewed as a dot product. There are s dot products in matrix-vector multiplication, and the operation count is

	M/D	A/S
Forming $\mathbf{A} \mathbf{x}$	s^2	$s^2 - s$

(d) Matrix-Matrix multiplication:

If $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{s \times s}$, where $\mathbf{A} = [a_{ij}]$ and $\mathbf{B} = [b_{ij}]$, then each entry in \mathbf{AB} , i.e., $(\mathbf{AB})_{ij} = \sum_{k=1}^s a_{ik} b_{kj}$, $i, j = 1, 2, \dots, s$, can be view as a dot product. There are s^2 entries in matrix-matrix multiplication, and the total counts are obtained as

	M/D	A/S
Forming \mathbf{AB}	s^3	$s^3 - s^2$

4.2 Operation counts of RK shape function and its derivatives

In section 2.1, we showed that for the moment $\mathbf{M}(x)$ matrix to be invertible in constructing the RK shape function, any position $x \in \Omega$ has to be covered by at least $n + 1$ kernel functions. Let κ be the overlapping number, we have the relationship

$$n + 1 \leq \kappa \ll Np \quad (4.1)$$

Before the construction of shape function, we need to form moment matrices, obtain the inversions, and form the correction functions. Let $s = n + 1$, then the vectors $\mathbf{H}(x - x_I)$ and $\mathbf{H}(0)$ have dimension s , the moment matrices have dimensions $s \times s$, and we the following operation counts:

Table 2 Operation counts for moment matrix and its derivatives

	M/D	A/S
Forming matrix $\mathbf{M}(x)$	$\kappa \cdot 2s^2$	$(\kappa - 1)s^2$
Forming matrix $\mathbf{M}_{,x}(x)$	$\kappa \cdot 6s^2$	$3(\kappa - 1)s^2 + 2$
Forming matrix $\mathbf{M}_{,xx}(x)$	$\kappa \cdot 12s^2 + 3s$	$6(\kappa - 1)s^2 + 5$

Table 3 Operation counts for inversions of moment matrices

	M/D	A/S
Finding inversion $\mathbf{M}^{-1}(x)$	s^3	$s^3 - 2s^2 + s$
Finding inversion $\mathbf{M}_{,x}^{-1}(x)$	$2s^3 + s^2$	$2(s^3 - s^2)$
Finding inversion $\mathbf{M}_{,xx}^{-1}(x)$	$3s^3 + 2s^2$	$3(s^3 - s^2) + 1$

Table 4 Operation counts for correction function and its derivatives

	M/D	A/S
Forming function $C(x)$	$s^2 + s$	$s^2 - 1$
Forming function $C_{,x}(x)$	$2(s^2 + s)$	$2(s^2 - 1) + 1$
Forming function $C_{,xx}(x)$	$3(s^2 + s) + 1$	$3(s^2 - 1) + 2$

Table 5 Operation counts for RK shape function and its derivatives

	M/D	A/S
Forming function $\psi_I(x)$	1	0
Forming function $\psi_{I,x}(x)$	2	1
Forming function $\psi_{I,xx}(x)$	3	2

For computing $\mathbf{M}_x^{-1}(x)$ and $\mathbf{M}_{,xx}^{-1}(x)$, the operation counts obtained following Eqs. (3.12) and (3.14) are given below:

After forming the moment matrices, we then construct the correction functions in Eqs. (3.4)-(3.6) and shape functions in Eqs. (3.1)-(3.3). The operation counts for these computations are given in Tables 4 and 5, respectively.

Consequently, the total operation counts for each RK shape function calculation at a give location are given as follows

$$\text{M/D: } s^3 + (2\kappa + 1)s^2 + s + 1 \quad (4.2)$$

$$\text{A/S: } s^3 + (\kappa - 2)s^2 + s - 1 \quad (4.3)$$

where $s \leq \kappa \ll Np$. The total operation counts for each first derivative $\psi_{I,x}(x)$ are

Table 6 Operation counts for RK shape functions with different reproducing degrees

	$\psi_I(x)$	$\psi_{I,x}(x)$	$\psi_{I,xx}(x)$
(M/D)	$3s^3 + s^2 + s + 1$	$11s^3 + 4s^2 + 3s + 2$	$26s^3 + 12s^2 + 6s + 4$
(A/S)	$2s^3 - 2s^2 + s - 1$	$7s^3 - 5s^2 + s + 1$	$16s^3 - 11s^2 + s + 12$
$n = 1$	31 (M/D)	112 (M/D)	272 (M/D)
$(s = 2)$	9 (A/S)	39 (A/S)	98 (A/S)
$n = 2$	94 (M/D)	344 (M/D)	832 (M/D)
$(s = 3)$	38 (A/S)	148 (A/S)	348 (A/S)
$n = 3$	213 (M/D)	782 (M/D)	1884 (M/D)
$(s = 4)$	99 (A/S)	373 (A/S)	864 (A/S)

$$\text{M/D: } 3s^3 + (8\kappa + 4)s^2 + 3s + 2 \quad (4.4)$$

$$\text{A/S: } 3s^3 + (4\kappa - 5)s^2 + s + 1 \quad (4.5)$$

For construction of a second derivative $\psi_{I,xx}(x)$, the operation counts are

$$\text{M/D: } 6s^3 + (20\kappa + 12)s^2 + 6s + 4 \quad (4.6)$$

$$\text{A/S: } 6s^3 + (10\kappa - 11)s^2 + s + 12 \quad (4.7)$$

In one-dimension, $s = n + 1$. For the case $s \approx \kappa$, the operation counts for the RK shape function and derivatives under different reproducing degree are listed in Table 6.

We can see from Table 6 that the operation counts for the construction of the first order derivative of RK shape function are about 4 times the counts for the construction of RK shape function, and the counts for construction of the second order derivative of RK shape function are about 9 times the counts for construction of RK shape function.

In general, the cost of forming the linear system under a strong formulation is still much less than that for a weak formulation since each non-zero entry of the matrix and vector of the weak formulation requires a domain integration.

4.3 Operation counts for solving a linear system

We take the second problem in Section 3.3, the Poisson problem with Dirichlet and Neumann boundary conditions, as an example for demonstration. From the collocation equations in Eqs. (3.28) and (3.29), we have

$$\sum_{I=1}^{Np} b_I \psi_{I,xx}(\xi_i) = f(\xi_i), \quad \forall \xi_i \in \Omega \quad (4.8)$$

$$\sum_{I=1}^{Np} b_I \psi_I(\xi_j) = g_D(\xi_j), \quad \xi_j \in \Gamma_0 \quad (4.9)$$

$$\sum_{I=1}^{Np} b_I \psi_{I,x}(\xi_k) = g_N(\xi_k), \quad \xi_k \in \Gamma_1 \quad (4.10)$$

where $\bar{\Omega} = \Omega \cup \Gamma_0 \cup \Gamma_1$, and Ns is total number of collocation points. We express the above

equations as

$$\mathbf{F}\mathbf{x} = \mathbf{r} \quad (4.11)$$

The operation counts for forming the matrix \mathbf{F} are given below:

$$\begin{aligned} \text{M/D: } & Np(s^3 + (2\kappa + 1)s^2 + s + 1) + Np(3s^3 + (8\kappa + 4)s^2 + 3s + 2) \\ & + Np(Ns - 2)(6s^3 + (20\kappa + 12)s^2 + 6s + 4) \end{aligned} \quad (4.12)$$

$$\begin{aligned} \text{A/S: } & Np(s^3 + (\kappa - 2)s^2 + s - 1) + Np(3s^2 + (4\kappa - 5)s^2 + s + 1) \\ & + Np(Ns - 2)(6s^3 + (10\kappa - 11)s^2 + s + 12) \end{aligned} \quad (4.13)$$

Besides, we need Ns functional evaluation in vector \mathbf{r} . Under the condition in Eq. (3.26), the number of collocation points can be chosen about two to four times the number of source points in computation. The system in Eq. (4.11) is over-determined, and we use following algorithm to find coefficient vector \mathbf{x} .

Algorithm based on Cholesky \mathbf{LL}^T :

- Step 1 Compute $\mathbf{F}^T\mathbf{F}$ and $\mathbf{z} = \mathbf{F}^T\mathbf{r}$
- Step 2 Cholesky decomposition on $\mathbf{F}^T\mathbf{F} = \mathbf{LL}^T$
- Step 3 Solve $\mathbf{Ly} = \mathbf{z}$ by backward substitution
- Step 4 Solve $\mathbf{L}^T\mathbf{x} = \mathbf{y}$ by forward substitution

The operation counts for obtaining the coefficient vector \mathbf{x} are given as follows

$$\text{M/D: } \frac{Ns \cdot Np^2}{2} + Ns \cdot Np + \frac{Np^3}{6} + \frac{Np^2}{2} + \frac{Np^2}{2} \quad (4.14)$$

4.4 Comparison of operation counts for weak and strong formulations

The total cost to obtain a solution by using RKCM consists of two parts: forming a linear system in Eq. (4.11) and the solution of the linear system.

Taking the Poisson problem discussed above as an example, we consider $Ns = 4 \times Np$ and $\kappa \approx s = n + 1$, where $n > 1$, then the operation count for forming a linear system Eq. (4.11) of RKCM is

$$\text{op: } 104Np^2 \cdot s^3 - 38Np \cdot s^3 \approx O(100s^3 \cdot Np^2) \quad (4.15)$$

The operation count for the solution of the linear system is

$$\text{op: } \frac{13}{6}Np^3 + 5Np^2 \approx O(2Np^3) \quad (4.16)$$

where op denotes one M/D plus one A/S, and $2 < s \ll Np$. In total, the operation count is proportional to Np^3 . For \mathbf{F} with a small dimension $Np \leq 50 \cdot s^3$, more CPU time is on forming the linear system. On the other hand, for \mathbf{F} with a large dimension $Np > 50 \cdot s^3$, most computation time are on solving the linear system with Cholesky decomposition.

Let's now consider the case for RKPM, the operation count for forming a linear system is

$$\text{op: } 11Np^2 \cdot s^3 + 6Np \cdot s^3 + \dots + Np^2 \text{ domain integrations} \quad (4.17)$$

Each domain integration consumes a much more CPU when higher order quadrature rules are used. If each domain integral consists of Np^2 local cell integration using Gaussian quadrature rules, the operation count in Eq. (4.17) becomes

$$\begin{aligned} \text{op: } 11Np^2 \cdot s^3 + 6Np \cdot s^3 + \dots + \mu Np^2 Np^2 \\ \approx O(10s^3 \cdot Np^2) + O(\mu \cdot Np^4) \end{aligned} \quad (4.18)$$

where the number μ is dependent on the order of Gaussian quadrature rules used. According to Eq. (4.18), CPU for forming a linear system is far beyond the CPU for solving the linear system. If Cholesky decomposition is used to solve the linear system of equations arising from RKPM (Steps 2~4 only), the operation count is

$$\text{op: } \frac{Np^3}{6} + \frac{Np^2}{2} + \frac{Np^2}{2} \approx O\left(\frac{1}{6}Np^3\right) \quad (4.19)$$

Finally we have a comparison of operation counts for RKPM and RKCM as

$$\underbrace{O(\mu \cdot Np^4) + O\left(\frac{1}{6}Np^3\right)}_{RKPM} > \underbrace{O(100s^3 \cdot Np^2) + O(2Np^3)}_{RKCM} \quad (4.20)$$

Overall, the total computational cost for weak formulation is higher than that for the strong formulation for larger systems.

5. Conclusions

The employment of reproducing kernel approximation under weak formulation (RKPM) has been widely used to solve PDEs, however, the method needs domain integration and special treatment of Dirichlet boundary conditions. Alternatively, non-local basis functions such as RBFs have been used as the approximation functions for solving PDEs under strong formulation, leading to a full and ill-conditioned system. To resolve these difficulties, we have introduced the compactly supported reproducing kernel approximation to the solution of PDFs under the strong collocation form (RKCM). This approach avoids domain integration and yields a banded and thus a much well-conditioned discrete system.

An important result in this study is to show that the degree of basis functions in the reproducing kernel approximation of RKCM has to be at least 2 for convergence. This is different from the case for RKPM, and the numerical experiments confirm this theoretical results. RKCM is as stable as the one based on weak formulation, such as FEM or RKPM. Further, the computational complexity of RKCM and RKPM are also analyzed and compared. By the operation counts of the two methods discussed in Section 4, we conclude that the RKCM is a simple and effective meshfree method for 2nd order PDEs that exist in many engineering and scientific problems.

Acknowledgements

The support for this work by National Science Council (Taiwan R.O.C.) is gratefully acknowledged.

References

- Ciarlet, P.G. (1978), *The finite Element Method for Elliptic Problem*, North-Holland, Inc., New York, NY.
- Belytschko, T., Krongauz, Y., Organ, D., Fleming, M. and Krysl, P. (1996), "Meshless methods: an overview and recent development", *Comput. Method. Appl. M.*, **139**, 3-49.
- Gingold, R.A. and Monaghan, J.J. (1977), "Smoothed particle hydrodynamics: theory and application to non-spherical stars", *Mon. Not. R. Astron. Soc.*, **181**, 375-389.
- Nayroles, B., Touzot, G. and Villon, P. (1992), "Generalizing the finite element method: diffuse approximation and diffuse elements", *Comput. Mech.*, **10**, 301-318.
- Belytschko, T., Lu, Y.Y. and Gu, L. (1994), "Element-free Galerkin methods", *Int. J. Numer. Meth. Eng.*, **37**, 229-256.
- Melenk, J.M. and Babuska, I. (1996), "The partition of unity finite element method: basic theory and applications", *Comput. Method. Appl. M.*, **139**, 289-314.
- Duarte, C.A. and Oden, J.T. (1996), "An h-p adaptive method using clouds", *Comput. Method. Appl. M.*, **139**, 237-262.
- Liu, W.K., Jun, S. and Zhang, Y.F. (1995), "Reproducing kernel particle methods", *Int. J. Numer. Meth. Fl.*, **20**, 1081-1106.
- Chen, J.S., Pan, C., Wu, C.T. and Liu, W.K. (1996), "Reproducing kernel particle methods for large deformation analysis of nonlinear structures", *Comput. Method. Appl. M.*, **139**, 195-227.
- Chen, J.S., Yoon, S., Wang, H.P. and Liu, W.K. (2000), "An improved reproducing kernel particle method for nearly incompressible hyperelastic solids", *Comput. Method. Appl. M.*, **181**, 117-145.
- Chen, J.S., Han, W., You, Y. and Meng, X. (2003), "A reproducing kernel method with nodal interpolation property", *Int. J. Numer. Meth. Eng.*, **56**, 935-960.
- Sukumar, N., Moran, B. and Belytschko, T. (1998), "The natural element method in solid mechanics", *Int. J. Numer. Meth. Eng.*, **43**, 839-887.
- Atluri, S.N. and Zhu, T.L. (2000) "The meshless local Petrov-Galerkin (MLPG) approach for solving problems in elasto-statics", *Comput. Mech.*, **25**, 169-179.
- Kansa, E.J. (1992a), "Multiquadrics - a scattered data approximation scheme with applications to computational fluid dynamics - I. Surface approximations and partial derivatives", *Comput. Math. Appl.*, **19**, 127-145.
- Kansa, E.J. (1992b), "Multiquadrics - a scattered data approximation scheme with applications to computational fluid dynamics - II. Solutions to parabolic, hyperbolic and elliptic partial differential equations", *Comput. Math. Appl.*, **19**, 147-161.
- Bernardi, C. and Maday, Y. (1997), *Handbook of Numerical Analysis, Techniques of Scientific Computing (Part 2)*, Vol. V, (Eds. Ciarlet, P.G. and Lions, J.L.), Elsevier Science Pub. Co., New York, NY.
- Han, W. and Meng, X. (2001), "Error analysis of the reproducing kernel particle method", *Comput. Method. Appl. M.*, **190**, 6157-6181.
- Hu, H.Y. and Li, Z.C. (2006), "Collocation methods for Poisson's equation", *Comput. Method. Appl. M.*, **195**, 4139-4160.
- Li, Z.C., Lu, T.T., Hu, H.Y. and Cheng, A.H.D. (2008), *Trefftz and Collocation Methods*, WIT press, Southampton, Boston.
- Hardy, R.L. (1971), "Multiquadric equations of topography and other irregular surfaces", *J. Geophys. Res.*, **176**, 1905-1915.
- Hardy, R.L. (1990), "Theory and applications of multiquadric-biharmonic method: 20 years of discovery", *Comput. Math. Appl.*, **19**, 163-208.
- Golub, G.H. and Van Loan, C.F. (1996), *Matrix Computations (3rd edition)*, The Johns Hopkins University Press, Baltimore and London.
- Hu, H.Y. and Chen, J.S. (2009), "An error analysis of collocation method based on reproducing kernel approximation", *Numer. Meth. Part. D. E.*, (in press).

- Burden, R.L. and Faires, J.D. (2005), *Numerical Analysis (8th edition)*, Thomson Brooks/Cole, Australia, Canada, Mexico, Singapore, Spain, UK, USA.
- Zhang, X., Chen, J.S. and Osher, S. (2008), "A multiple level set method for modeling grain boundary evolution of polycrystalline materials", *Interact. Mult. Mech.*, **1**, 178-191.
- Rojek, J. and Onate, E. (2008), "Multiscale analysis using a couple discrete/finite element model", *Interact. Mult. Mech.*, **1**, 1-31.

Appendix

Operation counts for solving fully dense and banded linear systems
Consider the following linear system of equations.

$$\begin{aligned} E_1: a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &= a_{1,n+1} \\ E_2: a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &= a_{2,n+1} \\ &\vdots \\ E_n: a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n &= a_{n,n+1} \end{aligned}$$

(a) Solution of a fully dense system (Algorithm A)

The solution procedures of a $n \times n$ linear system by using Gaussian Elimination with Backward Substitution are summarized as follows.

INPUT Number of unknowns and equations n ; augmented matrix $A = [a_{ij}]$, where $1 \leq i \leq n$ and $1 \leq j \leq n+1$

OUTPUT Solution x_1, x_2, \dots, x_n or a message that the linear system has no unique solution

Step 1: For $i = 1, 2, \dots, n$ do steps 2~4

Step 2: Let p be the smallest integer with $i \leq p \leq n$ and $a_{pi} \neq 0$. If no integer p can be found, then

OUTPUT ("no unique solution exists"); STOP

Step 3: If $p \neq i$, then perform $(E_p) \leftrightarrow (E_i)$

Step 4: For $j = i+1, i+2, \dots, n$ do steps 5~6

Step 5: Set $m_{ji} = a_{ji}/a_{ii}$

Step 6: Perform $(E_j - m_{ji}E_i) \rightarrow (E_j)$;

Step 7: If $a_{nn} = 0$, then OUTPUT ("no unique solution exists"); STOP

Step 8: Set $x_n = a_{n,n+1}/a_{nn}$

Step 9: For $i = n-1, n-2, \dots, 1$ set $x_i = \left[a_{i,n+1} - \sum_{j=i+1}^n a_{ij}x_j \right] / a_{ii}$

Step10: OUTPUT (x_1, x_2, \dots, x_n) ; STOP

In the above, Step 5 requires $n-i$ division operations. In Step 6, the replacement of E_j by $(E_j - m_{ji}E_i)$ requires $(n-i)(n-i+1)$ multiplications. Thereafter, each term of equations requires $(n-i)(n-i+1)$ subtractions. For each counter $i = 1, 2, \dots, n-1$, the subtotal operations required in Step 5 and Step 6 are

$$M/D: (n-i) + (n-i)(n-i+1) = (n-i)(n-i+2)$$

$$A/S: (n-i)(n-i+1)$$

Summing up the counts for each i , we have

$$M/D: \sum_{i=1}^{n-1} (n-i)(n-i+2) = \frac{2n^3 + 3n^2 - 5n}{6}$$

$$A/S: \sum_{i=1}^{n-1} (n-i)(n-i+1) = \frac{n^3 - n}{3}$$

In addition, step 8 requires 1 division, and Step 9 requires $n-i$ multiplications and $n-i-1$ additions. The numbers of these operations are

$$M/D: 1 + \sum_{i=1}^{n-1} ((n-i) + 1) = \frac{n^2 + n}{2}$$

$$A/S: \sum_{i=1}^{n-1} ((n-i-1) + 1) = \frac{n^2 - n}{2}$$

The total amount of arithmetic operations required in *Algorithm A* are

$$M/D: \frac{2n^3 + 3n^2 - 5n}{6} + \frac{n^2 + n}{2} = \frac{n^3}{3} + n^2 - \frac{n}{3} \quad (A.1)$$

$$A/S: \frac{n^3 - n}{3} + \frac{n^2 - n}{2} = \frac{n^3}{3} + \frac{n^2}{2} - \frac{5n}{6} \quad (A.2)$$

For large n , M/D and A/S are in the order of $\frac{n^3}{3}$.

(b) Solution of a banded system (Algorithm B)

The solution procedures of a $n \times n$ linear system with bandwidth $2w-1$ using Gaussian Elimination with Backward Substitution are described as follows.

INPUT Number of unknowns and equations n ; augmented matrix $A = [a_{ij}]$, where $1 \leq j \leq n$ and $1 \leq i \leq n+1$

OUTPUT Solution x_1, x_2, \dots, x_n or a message that the linear system has no unique solution

Step 1: For $i = 1, 2, \dots, n$ do steps 2~4

Step 2: Let p be the smallest integer with $i \leq p \leq n$ and $a_{pi} \neq 0$. If no integer p can be found, then

OUTPUT ("no unique solution exists"); STOP

Step 3: If $p \neq i$, then perform $(E_p) \leftrightarrow (E_i)$

Step 4: For $j = i+1, i+2, \dots, i+w-1$ do steps 5~6 (Note that $i+w-1 \leq n$)

Step 5: Set $m_{ji} = a_{ji}/a_{ii}$

Step 6: Perform $(E_j - m_{ji} E_i) \rightarrow (E_j)$

Step 7: If $a_{nn} = 0$, then OUTPUT ("no unique solution exists"); STOP

Step 8: Set $x_n = a_{n,n+1}/a_{nn}$

Step 9: For $i = n-1, n-2, \dots, 1$ set $x_i = \left[a_{i,n+1} - \sum_{j=i+1}^{i+w-1} a_{ij}x_j \right] / a_{ii}$

Step10: OUTPUT (x_1, x_2, \dots, x_n) ; STOP

In the above, Step 5 requires $w-1$ (for $i \leq n-w+1$) or $n-i$ (for $i \geq n-w+2$) division operations. In Step 6, the replacement of E_j by $(E_j - m_{ji}E_i)$ requires $(w-1)w$ (for $i \leq n-w+1$) or $(n-i)(n-i+1)$ (for $i \geq n-w+2$) multiplications and $(w-1)w$ (for $i \leq n-w+1$) or $(n-i)(n-i+1)$ (for $i \geq n-w+2$) subtractions. For each counter $i = 1, 2, \dots, n-1$, the operations required in Step 5 and Step 6 are

$$\begin{aligned} \text{M/D: } & \begin{cases} (w-1) + (w-1)w = (w-1)(w+1) = w^2-1, & \text{for } i \leq n-w+1 \\ (n-i) + (n-i)(n-i+1) = (n-i)(n-i+2), & \text{for } i \geq n-w+2 \end{cases} \\ \text{A/S: } & \begin{cases} (w-1)w, & \text{for } i \leq n-w+1 \\ (n-i)(n-i+1), & \text{for } i \geq n-w+2 \end{cases} \end{aligned}$$

Summing up the counts for each i , we have

$$\begin{aligned} \sum_{i=1}^{n-w+1} (w^2-1) + \sum_{i=n-w+2}^{n-1} (n-i)(n-i+2) &= n(w^2-1) - \frac{2w^3}{3} + \frac{w^2}{2} + \frac{w}{6} \\ \sum_{i=1}^{n-w+1} (w^2-w) + \sum_{i=n-w+2}^{n-1} (n-i)(n-i+1) &= n(w^2-w) - \frac{2w^3}{3} + w^2 - \frac{w}{3} \end{aligned}$$

In addition, Step 8 requires 1 division, and in Step 9 requires $w-1$ (for $i \leq n-w+1$) or $n-i$ (for $i \geq n-w+2$) multiplications and $w-2$ (for $i \leq n-w+1$) or $n-i-1$ (for $i \geq n-w+2$) additions. The numbers of these operations are

$$\begin{aligned} \text{M/D: } & 1 + \sum_{i=1}^{n-w+1} w + \sum_{i=n-w+2}^{n-1} ((n-i)+1) = nw - \frac{w^2}{2} + \frac{w}{2} \\ \text{A/S: } & \sum_{i=1}^{n-w+1} (w-1) + \sum_{i=n-w+2}^{n-1} ((n-i-1)+1) = n(w-1) - \frac{w^2}{2} + \frac{w}{2} \end{aligned}$$

The total amount of arithmetic operations required in *Algorithm B* are

$$\text{M/D: } n(w^2 + w - 1) - \frac{2w^3}{3} + \frac{2w}{3} \quad (\text{A.3})$$

$$\text{A/S: } n(w^2 - 1) - \frac{2w^3}{3} + \frac{w^2}{2} + \frac{w}{6} \quad (\text{A.4})$$

For large n , M/D is approximately $n(w^2 + w - 1)$ and A/S is approximately $n(w^2 - 1)$. Since for a sparse and banded linear system, $w \ll n$, it is more effective to use *Algorithm B*.

For a special case in *Algorithm B* by letting $w = n$, we recover the operation counts of *Algorithm A*:

$$(\text{A.3}) = n(n^2 + n - 1) - \frac{2n^3}{3} + \frac{2n}{3} = \frac{n^3}{3} + n^2 - \frac{n}{3} = (\text{A.1})$$

$$(\text{A.4}) = n(n^2 - 1) - \frac{2n^3}{3} + \frac{n^2}{2} + \frac{n}{6} = \frac{n^3}{3} + \frac{n^2}{2} - \frac{5n}{6} = (\text{A.2})$$