Accurately solving the Poisson equation by combining multiscale radial basis functions and Gaussian quadrature

Zhenyu Liu, J. G. Korvink

INTEK-Institute for Microsystems Technology, Albert Ludwig University, Freiburg, Germany

Abstract

We propose a method combining multiscale radial basis functions and Gaussian quadrature that greatly improves the numerical solution accuracy when applied to the Poisson equation defined over arbitrary-shaped domains. Instead of directly using the collocation points as interpolation points, the Gaussian quadrature points are used as the interpolation points in the discretised domain. In combining the adaptive positioning of multiscale Wendland radial base functions and the symmetric collocation method, our numerical examples demonstrate that this method improves the accuracy of solution significantly, while the computational cost is no big increase and of the same order when compared with the “standard” symmetric method.

1 Introduction

Recently, meshless methods as applied to the numerical solution of partial differential equations (PDEs) have received much attention. To date, several kinds of meshless methods have been proposed, and include the element-free Galerkin method [1], the reproducing kernel method [2], h-p clouds [3] and the partition of unity method [4].

The radial basis function (RBF) method was first used as an effective approximation technique for multivariate scattered data and for function interpolation. The RBF approach is an inherently meshless method when combined with the collocation method. The first report to apply RBFs to solve a PDE appeared in 1990 [5]. Initially, a nonsymmetric method was used to discretise the PDE, but now we know that sometimes the invertibility of the resulting numerical matrix is not guaranteed, even though pathological cases seem to be very rare [6]. One paper [7] views the solution of a PDE as a special case of Hermite-Birkhoff interpolation. The collocation matrix that arises from using a Hermite-Birkhoff formulation is positive defi-
nite, hence the resulting numerical equation is (at least in principle) always solvable. Hermite-Birkhoff interpolation is often referred to as the symmetric method. Compared with a nonsymmetric method, the symmetric method satisfies all the boundary conditions and provides for higher accuracy of the numerical solution, even though an extra step is needed to construct the collocation matrix.

When using symmetric RBFs together with the collocation method to solve a PDE, the interpolation points are often chosen to be the same as the collocation points. This means that the partial differential operator is satisfied exactly on the collocation points, but not over the whole discretised domain. The multilevel and smoothing method [8] and the least-squares method with auxiliary points [9] are proposals aimed to improve the accuracy of numerical solution.

In this paper, multiscale compactly supported Wendland radial functions are chosen as basis functions with which to construct the approximate solution of a PDE, and Gaussian quadrature together with the symmetric Hermite-Birkhoff method is used to produce a discrete equation system that corresponds to the partial differential equation.

We follow a different approach when our method is compared to the direct symmetric collocation method. The position of collocation points and the support size of each radial base is adapted according to the gradient change of the local numerical solution. At the same time, the interpolation points are not limited to coincide with the collocation points. Additional Gaussian quadrature points are placed around each collocation point and are used to interpolate the solution. In this way the partial differential operator is satisfied over the whole discretised domain within the broader meaning of quadrature. The dimension of the resulting numerical matrix is exactly the same as for a non-modified symmetric method, whereas the solution accuracy is improved significantly. The numerical results presented in section 4 clearly demonstrate the effectiveness of the proposed method.

2 Solving PDEs with the symmetric RBF method

There are two different approaches used when formulating the RBF expansion for the collocation solution of a PDE which has the form

\[
\begin{align*}
LU &= f \text{ in } \Omega \\
BU &= g \text{ on } \partial \Omega
\end{align*}
\]

where \( L \) and \( B \) are differential operators and \( U \) is the solution. The first approach is the so-called non-symmetric method. The approximation \( u \) of the solution is constructed as

\[
u(x) = \sum_{i=1}^{n} C_i \Phi(\|x - x_i\|_2) \quad x \in \mathbb{R}^d
\]
where $\Phi$ is an RBF and $x_i$ denotes a collocation point. After collocation we obtain the following linear equation system

$$
\begin{bmatrix}
L \Phi \\
B \Phi
\end{bmatrix} C =
\begin{bmatrix}
f \\
0
\end{bmatrix}
$$

(3)

The second approach is the symmetric method which is motivated by Hermite-Birkhoff interpolation. The approximation $u$ of the solution is constructed as

$$u(x) = \sum_{i=1}^{n} C_i \Phi(\|x - x_i\|_2) + \sum_{j=1}^{n_t} D_j \Phi(\|x - x_j\|_2) \quad x \in \mathbb{R}^d$$

(4)

where $n$ denotes the number of collocation points in the domain and $n_t$ denotes the number of collocation points on the boundary. After collocation we obtain the following linear equation system

$$
\begin{bmatrix}
LL \Phi & LB \Phi \\
BL \Phi & BB \Phi
\end{bmatrix}
\begin{bmatrix}
C \\
D
\end{bmatrix} =
\begin{bmatrix}
f \\
g
\end{bmatrix}
$$

(5)

Due to the connection of the symmetric method to Hermite-Birkhoff interpolation, the matrix formulation is well-posed and we obtain a positive-definite symmetric system. For smooth problems (for example, when the solution of the PDE is smooth enough), the accuracy of the symmetric method is higher than for the unsymmetric method. However, when the solution is not smooth enough in the computational domain (for example, the electrostatic problem), one cannot say the symmetric method must better than the unsymmetric method.

In addition, a convergence proof exists for linear elliptic PDE discretised using Wendland compactly-supported RBFs in conjunction with the symmetric collocation method [11].

3 Multiscale expansion of an approximate solution and Gaussian quadrature

When radial basis functions were first applied to the numerical solution of PDEs, they were globally supported RBFs, and were in the form of multi-quartics (In 1990 Kansa first uses multi-quartics RBF to solve the PDE) or thin-plate splines. It is well known that globally supported RBFs lead to dense numerical matrices. For large-scale simulations, these may also be the cause of numerical problems in the linear equation solver, with symptoms such as a slow convergence rate and perhaps even an instable iterative process.

Compactly supported RBFs provide a way to obtain a sparse matrix. The Wendland basis [10] is one of the most popular classes of compactly supported RBFs. The definition of the Wendland basis $\psi_{l,k}$ is

$$\psi_{l,k} = I^k \psi_{l,0}(r)$$

(6)

where $\psi_{l,0}$ is a truncated power expansion of order $l$ and $I$ is an integral operator of order $k$. Even though the Wendland functions could be computed recursively,
they can also be expressed as polynomials. These have only one parameter and are
strictly positive definite. For example, for $\psi_{5,3}$, we get

$$\psi_{5,3}(r) = (1 - r)^8(32r^2 + 25r + 8r + 1)$$

which is in $C^6$ and is strictly positive definite in $R^3$.

The Wendland basis is compactly supported in $r \in [-1, 1]$. Nevertheless, the
actual support size can be scaled to be proportional to the distance between collocation
points in the neighbourhood of the center of the basis. For example

$$\psi_{5,3}(r, h) = (1 - r/h)^8(32(r/h)^2 + 25(r/h)^2 + 8r/h + 1)$$

in which $h$ is the support size. So the functions appear to be similar to wavelets,
which are able to define a dilation and translation, but lack the wavelet’s orthogonal
property. Therefore, approximate solutions construction with a Wendland basis be-
long to the set of nonorthogonal function expansions.

It is well-known that the success of the collocation method depends strongly on
the selection of the basis functions and on the positions of the collocation points.
Scaling is an important technique with which to improve the approximation of an
analytical solution. We do this with a linear combination of functions defined
through the translation and dilation of one parent Wendland basis. The positioning
of collocation points also can affect the convergence of the interpolation scheme.
Generally, we spread the collocation points equally in the design domain for con-
venience, or use the quadrature points of the design domain as collocation points to
obtain higher accuracy. These methods do not consider the character of the PDE so-
lution when setting up the collocation points. In fact, we could use fairly flat basis
functions with large support radius in smooth areas of the solution and sharp basis
functions with small support radius in areas where the solution shape changes
quickly.

In this paper, a hierarchical set of collocation points is proposed to approximate
the numerical solution. First, the collocation points are placed on a regular grid with
a uniform support radius for which we compute a coarse approximation of the so-
lution. At the next level, in a refinement step, the bases close to the positions where
the solution has large gradients are replaced by shifted and compressed basis func-
tions. Figure 1 shows a hierachic expansion of basis function space, at the first
(Figure 1b), second (Figure 1c) and third refinement level (Figure 1d). In a refine-
ment step, the support radius of the new basis functions are halved. Here the refine-
ment is part of a 1-D analysis for interpolating the function of Figure 1a, but also
see Example 1.

Usually the interpolation points are selected to coincide with the collocation
points of the collocation method. In this case the partial differential operator is sat-
sified exactly at the collocation points. This is rather restrictive, and in fact any
point in design domain should be available as an interpolation point.

To improve the accuracy of the solution, we could of course use more collocation
points, but this will increase the computational cost. At the same time, too
many collocation points will also increase the conditional number of the resulting
Figure 1: Hierachic basis functions for interpolation. a) The 1D analytical solution of example 1. This is the curve that the numerical process converges towards. b) Six equally-spaced Wendland RBFs. c) Function 4 is adapted to form 4a and 4b. d) Function 4b is adapted to form 4ba and 4bb. Note how the adapted functions become less spread (smaller support) and taller (to maintain normality, equation (11)).

matrix. The alternative is choosing an optimal distribution of collocation points. Using Gauss-Legendre points for collocation is well known in the orthogonal collocation and spline collocation methods [12]. But the location of collocation points is always fixed in the discretised domain if optimal accuracy is required.

The quadrature method is a general procedure for solving a PDE in the sense of approximation. For the quadrature method, we consider not only the partial differential equation

$$LU = f$$

in a collocation point, but also consider the immediate vicinity around the collocation point. We express the equilibrium condition around the point as

$$\int_{\Omega_i} (LU - f) d\Omega_i = 0$$

The key then is how to choose the quadrature points to minimize the error. We chose Gaussian quadrature for accuracy, because compactly supported Wendland RBFs are higher order polynomials, for which Gaussian quadrature is known to be good.
4 Numerical examples

In this section, we consider two Poisson equations, one in 1D and the other in 2D, whose analytical solution is nevertheless readily available and therefore can easily be compared with our computed numerical solution. In all of the following examples we use CS-Wendland RBFs $\psi(x)$ which satisfy the normal condition

$$\int_{\Omega} \psi d\Omega = 1$$

and hence ensure consistency of the resulting numerical solutions.

Example 1: 1D Poisson equation

We consider a 1D Poisson equation along the $x$-axis

$$\nabla^2 u(x) = 2 - 12x^2 \quad x \in [0, 1]$$

subject to the Dirichlet boundary conditions

$$u(0) = u(1) = 0$$

We select a uniform grid of 11 nodes with the same support size of 1.0 at the initial step. The bases in the positions $x = 0.7$, 0.8 and 0.9 are replaced by new bases with a smaller support size of 0.6 at the subsequent step. The $a \ posteriori$ refinement decision is based on the coarse approximate solution of the preceding step. The quadrature method is used to iteratively improve the accuracy of the coarse solution. Two different kinds of quadrature methods, Gaussian quadrature and Newton-Cotes quadrature, are used. Table 1 shows the $L_2$ error of the approximate numerical solution relative to the exact solution $u(x) = x^2(1 - x^2)$.

The column marked sym1 contains results from the solution of equations (12) and (13) by the symmetry method using the collocation points of the first step, and sym2 using the collocation points of the second step. The quar2 and quar3 columns consider solution by the quadrature method with two and three quadrature points in the second step. Table 2 shows, for the above cases, the relative error $|u - U|/u$ at the maximal value point $x = \sqrt{2}/2$, i.e., where $\partial u/\partial x = 0 = 2x(1 - 2x^2)$.

Example 2: 2D Poisson equation

We consider a 2D Poisson equation over the $xy$-plane

$$\left(\nabla^2 u(x, y) = -30000e^{1000(x^6 + y^6)}(x^4 + y^4 - 1200(x^{10} + y^{10}))\right)$$

for $(x, y) \in [-0.5, 0.5]$)

subject to the Dirichlet boundary conditions

$$u(x, y) = e^{-1000(x^6 + y^6)}$$

on $x = -0.5, x = 0.5, y = -0.5, y = 0.5$

We select a uniform grid $11 \times 11$ of nodes with an identical support size of 1.0 at the initial solution step. The bases at the positions where $x$ and $y$ are equal to
and quadr3 have the same meaning as for example l.

Table 1: L_2 error of the numerically computed solution for the 1D Poisson equation of example 1.

<table>
<thead>
<tr>
<th>N</th>
<th>sym1</th>
<th>sym2</th>
<th>quar2</th>
<th>quar3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Newton-Cotes</td>
<td>4.329x10^{-2}</td>
<td>2.943x10^{-2}</td>
<td>2.838x10^{-2}</td>
<td>1.293x10^{-2}</td>
</tr>
<tr>
<td>Gaussian</td>
<td>4.329x10^{-2}</td>
<td>2.943x10^{-2}</td>
<td>9.574x10^{-3}</td>
<td>1.377x10^{-3}</td>
</tr>
</tbody>
</table>

Table 2: Relative error example 1 at the maximal value point x = \sqrt{2}/2.

<table>
<thead>
<tr>
<th>N</th>
<th>sym1</th>
<th>sym2</th>
<th>quar2</th>
<th>quar3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Newton-Cotes</td>
<td>6.787 %</td>
<td>3.503 %</td>
<td>3.354 %</td>
<td>1.325 %</td>
</tr>
</tbody>
</table>

Figure 2: Plot of the solution of example 2. a) Exact analytical solution u(x, y) = exp[-1000(x^6 + y^6)]. b) Solution for the symmetry method sym2 of Table 3. c) Solution for the quadrature method quar3 of Table 3, showing a dramatic improvement in quality.

5 Summary

A gaussian quadrature method based on multiscale RBFs and symmetric collocation is proposed in this paper. The total number of unknowns in this method is equal to the symmetric RBF method, but the accuracy of numerical solution is significantly improved. Numerical examples show that the new method is well-suited to solving the Poisson equation which has smooth solution.
Table 2: Relative error example 1 at the maximal value point $x = \sqrt{2}/2$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$\text{sym1}$</th>
<th>$\text{sym2}$</th>
<th>$\text{quar2}$</th>
<th>$\text{quar3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gaussian</td>
<td>6.787%</td>
<td>3.503%</td>
<td>1.179%</td>
<td>0.1072%</td>
</tr>
</tbody>
</table>

Table 3: $L_2$ error of the numerically computed solution for the 2D Poisson equation of example 2.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$\text{sym1}$</th>
<th>$\text{sym2}$</th>
<th>$\text{quar2}$</th>
<th>$\text{quar3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_2$ Error</td>
<td>3.717x10^{-1}</td>
<td>1.182x10^{-1}</td>
<td>6.097x10^{-2}</td>
<td>6.347x10^{-3}</td>
</tr>
</tbody>
</table>

References