

RBF Collocation Methods and Pseudospectral Methods

G. E. Fasshauer*

Draft: November 19, 2004

Abstract

We show how the collocation framework that is prevalent in the radial basis function literature can be modified so that the methods can be interpreted in the framework of standard pseudospectral methods. This implies that many of the standard algorithms and strategies used for solving time-dependent as well as time-independent partial differential equations with (polynomial) pseudospectral methods can be readily adapted for the use with radial basis functions. The potential advantage of radial basis functions is that they lend themselves to complex geometries and non-uniform discretizations.

Keywords: radial basis functions, collocation, pseudospectral methods.

1 Introduction

Pseudospectral (PS) methods are known as highly accurate solvers for partial differential equations (PDEs). The basic idea (see, e.g., [6] or [17]) is to use a set of (very smooth and global) basis functions ϕ_j , $j = 1, \dots, N$, such as polynomials to represent an unknown function (the approximate solution of the PDE) via

$$u^h(x) = \sum_{j=1}^N \lambda_j \phi_j(x), \quad x \in \mathbb{R}. \quad (1)$$

Since most of our discussion will focus on a representation of the spatial part of the solution we ignore the time variable in the formulas for u^h . We will employ standard time-stepping procedures to deal with the temporal part of the solution. Moreover, since standard pseudospectral methods are designed for the univariate case we initially limit ourselves to single-variable functions. Later we will generalize to multivariate (spatial) problems by using radial basis functions.

An important feature of pseudospectral methods is the fact that one usually is content with obtaining an approximation to the solution on a discrete set of grid points x_i , $i = 1, \dots, N$. One of several ways to implement the spectral method is via so-called *differentiation matrices*, i.e., one finds a matrix D such that at the grid points x_i we have

$$\mathbf{u}' = D\mathbf{u}, \quad (2)$$

*Department of Applied Mathematics, Illinois Institute of Technology, Chicago, IL 60616, USA

where $\mathbf{u} = [u^h(x_1), \dots, u^h(x_N)]^T$ is the vector of values of u^h at the grid points. Frequently, orthogonal polynomials such as Chebyshev polynomials are used as basis functions, and the grid points are corresponding Chebyshev points. In this case the entries of the differentiation matrix are explicitly known (see, e.g., [17]).

In this paper we are interested in using (infinitely smooth) radial basis functions (RBFs) in the spectral expansion (1), i.e., $\phi_j(x) = \Phi(\|x - x_j\|)$, where Φ is some positive definite univariate basic function. Popular choices for positive definite functions include inverse multiquadrics

$$\Phi(r) = \frac{1}{\sqrt{1 + (\varepsilon r)^2}}, \quad (3)$$

Gaussians

$$\Phi(r) = e^{-(\varepsilon r)^2}, \quad (4)$$

Matérn functions (see, e.g., [1]) such as

$$\Phi(r) = e^{-\varepsilon r} ((\varepsilon r)^3 + 6(\varepsilon r)^2 + 15(\varepsilon r) + 15), \quad (5)$$

or compactly supported Wendland functions (which do not lend themselves to the task at hand, i.e., as generalized PS methods, since they are of limited smoothness). With some additional notational effort all that follows can also be formulated for conditionally positive definite functions such as the popular multiquadric

$$\Phi(r) = \sqrt{1 + (\varepsilon r)^2}. \quad (6)$$

Above, the univariate variable r is a radial variable, i.e., $r = \|x\|$, and the positive parameter ε is equivalent to the well-known shape parameter used to scale the basic functions. We have chosen the representations above since then $\varepsilon \rightarrow 0$ always results in “flat” basic functions for which we have the well-known trade-off principle, i.e., high accuracy at the cost of low stability or vice versa (see, e.g., [15]).

2 Differentiation Matrices

We now begin with a general discussion of differentiation matrices. Consider expansion (1) and let ϕ_j , $j = 1, \dots, N$, be an arbitrary linearly independent set of smooth functions that will serve as the basis for our approximation space.

If we evaluate (1) at the grid points x_i , $i = 1, \dots, N$, then we get

$$u^h(x_i) = \sum_{j=1}^N \lambda_j \phi_j(x_i), \quad i = 1, \dots, N,$$

or in matrix-vector notation

$$\mathbf{u} = A\boldsymbol{\lambda}, \quad (7)$$

where $\boldsymbol{\lambda} = [\lambda_1, \dots, \lambda_N]^T$ is the coefficient vector, the evaluation matrix A has entries $A_{ij} = \phi_j(x_i)$, and \mathbf{u} is as before.

By linearity we can also use the expansion (1) to compute the derivative of u^h by differentiating the basis functions

$$\frac{d}{dx} u^h(x) = \sum_{j=1}^N \lambda_j \frac{d}{dx} \phi_j(x).$$

If we again evaluate at the grid points x_i then we get in matrix-vector notation

$$\mathbf{u}' = A_x \boldsymbol{\lambda}, \quad (8)$$

where \mathbf{u} and $\boldsymbol{\lambda}$ are as above, and the matrix A_x has entries $\frac{d}{dx} \phi_j(x_i)$, or, in the case of radial functions, $\frac{d}{dx} \Phi(\|x - x_j\|)|_{x=x_i}$.

In order to obtain the differentiation matrix D we need to ensure invertibility of the evaluation matrix A . This depends both on the basis functions chosen as well as the location of the grid points x_i . For univariate polynomials it is well-known that the evaluation matrix is invertible for any set of distinct grid points. In particular, if the polynomials are written in cardinal (or Lagrange) form, then the evaluation matrix is the identity matrix. For positive definite radial basis functions (an extension of) Bochner's theorem guarantees the invertibility of the matrix A for any set of distinct grid points (also non-uniformly spaced and in \mathbb{R}^d , $d > 1$). Cardinal RBFs, on the other hand, are rather difficult to obtain. For the special case of uniform one-dimensional grids such formulas can be found in [13].

Thus we can use (7) to solve for the coefficient vector $\boldsymbol{\lambda} = A^{-1} \mathbf{u}$, and then (8) yields

$$\mathbf{u}' = A_x A^{-1} \mathbf{u},$$

so that the differentiation matrix D corresponding to (2) is given by

$$D = A_x A^{-1}.$$

For more complex linear differential operators \mathcal{L} with constant coefficients we can use the same argument as above to obtain a discretized differential operator (differentiation matrix)

$$L = A_{\mathcal{L}} A^{-1}, \quad (9)$$

where the matrix $A_{\mathcal{L}}$ has entries $A_{\mathcal{L},ij} = \mathcal{L} \phi_j(x_i)$. In the case of radial basis functions these entries are of the form $A_{\mathcal{L},ij} = \mathcal{L} \Phi(\|x - x_j\|)|_{x=x_i}$.

In the context of spectral methods the differentiation matrix L can now be used to solve all kinds of PDEs (time-dependent as well as time-independent). Sometimes only multiplication by L is required (e.g., for many time-stepping algorithms), and for other problems one needs to be able to invert L . In the standard PS case it is known that the Chebyshev differentiation matrix has an N -fold zero eigenvalue (see [3], p.70), and thus is not invertible by itself. However, once boundary conditions are taken into consideration the situation changes (see, e.g., [17], p.67).

To obtain a little more insight into the special properties of radial basis functions let us pretend to solve the (ill-posed) linear PDE of the form $\mathcal{L}u = f$ by ignoring boundary conditions. An approximate solution at the grid points x_i might be obtained by solving the discrete linear system

$$L\mathbf{u} = \mathbf{f},$$

where \mathbf{f} contains the values of f at the grid points and L is as above. In other words, the solution at the grid points is given (see (9)) by

$$\mathbf{u} = L^{-1} \mathbf{f} = A(A_{\mathcal{L}})^{-1} \mathbf{f},$$

and we see that invertibility of L (and therefore $A_{\mathcal{L}}$) would be required.

As mentioned above, the differentiation matrix for pseudospectral methods based on Chebyshev polynomials is singular. This is only natural since the problem of reconstructing an unknown function from the values of its derivative alone is ill-posed.

However, if we use radial basis functions it is well known in the RBF literature that the matrix $A_{\mathcal{L}}$ is invertible provided a positive definite basic function is used, the differential operator is elliptic and no boundary conditions are present. Therefore, the basic differentiation matrix L for RBF-based pseudospectral methods is invertible.

The observation just made suggests that RBF methods are sometimes “too good to be true”. They may deliver a “solution” even for ill-posed problems. This is a consequence of the variational framework in which the RBF method can be formulated (see, e.g., [18]), i.e., RBF methods possess a built-in regularization capability.

3 PDEs with Boundary Conditions via Pseudospectral Methods

First we discuss how the linear elliptic PDE problem

$$\mathcal{L}u = f \quad \text{in } \Omega$$

with Dirichlet boundary condition

$$u = g \quad \text{on } \Gamma = \partial\Omega$$

can be solved using spectral methods if the basis functions do not already satisfy the boundary conditions (see, e.g., [17], Program 36). Essentially, one starts with the differentiation matrix L based on all grid points x_i , and then replaces the diagonal entries corresponding to boundary points with ones and the remainder of those rows with zeros. This corresponds to enforcing the boundary condition $u = g$ explicitly. By reordering the rows and columns of the resulting matrix we obtain a block matrix of the form

$$L_{\Gamma} = \begin{bmatrix} M & P \\ 0 & I \end{bmatrix}, \quad (10)$$

where the non-zero blocks M and I are of size $(N - N_B) \times (N - N_B)$ and $N_B \times N_B$, respectively, and N_B denotes the number of grid points on the boundary Γ .

The solution of the PDE with boundary conditions on the grid is then given by the solution of the block linear system

$$L_{\Gamma}\mathbf{u} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix}, \quad (11)$$

where the vectors \mathbf{f} and \mathbf{g} collect the values of f and g at the respective grid points.

We can decompose the vector of grid values of the solution into $\mathbf{u} = [\mathbf{u}_{\Omega}, \mathbf{u}_{\Gamma}]^T$, where \mathbf{u}_{Ω} collects the values in the interior of the domain Ω and \mathbf{u}_{Γ} collects the values on the boundary. Solving (11) for $\mathbf{u}_{\Gamma} = \mathbf{g}$ and substituting this back in we obtain

$$\mathbf{u}_{\Omega} = M^{-1}(\mathbf{f} - P\mathbf{g}),$$

or, for homogeneous boundary conditions,

$$\mathbf{u}_\Omega = M^{-1} \mathbf{f}.$$

We now see that we need to be able to decide whether the matrix M is invertible. In the case of Chebyshev polynomial basis functions and the second-derivative operator $\frac{d^2}{dx^2}$ coupled with different types of boundary conditions this question has been answered affirmatively by Gottlieb and Lustman ([8], or, e.g., Section 11.4 of [3]). Program 15 of [17] also provides a discussion and an illustration of one such problem.

4 PDEs with Boundary Conditions via RBFs

Once boundary conditions are added to the PDE $\mathcal{L}u = f$ then two collocation approaches are commonly used in the RBF community. For the sake of simplicity we restrict our discussion to Dirichlet boundary conditions.

4.1 Kansa's Non-symmetric Collocation Method

In Kansa's non-symmetric method [11] one starts with the expansion

$$u^h(x) = \sum_{j=1}^N \lambda_j \phi_j(x), \quad x \in \Omega \subseteq \mathbb{R}^d, \quad (12)$$

just as before (cf. (1)). However, the coefficient vector $\boldsymbol{\lambda}$ is now determined by inserting (12) into the PDE and boundary conditions and forcing these equations to be satisfied at the grid points x_i . The *collocation solution* is therefore obtained by solving the linear system

$$\begin{bmatrix} \tilde{A}_{\mathcal{L}} \\ \tilde{A} \end{bmatrix} \boldsymbol{\lambda} = \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix}, \quad (13)$$

where \mathbf{f} and \mathbf{g} are as above, and the (rectangular) matrices $\tilde{A}_{\mathcal{L}}$ and \tilde{A} are of the form

$$\begin{aligned} \tilde{A}_{\mathcal{L},ij} &= \mathcal{L}\phi_j(x_i) = \mathcal{L}\Phi(\|x - x_j\|)|_{x=x_i}, \quad i = 1, \dots, N - N_B, \quad j = 1, \dots, N, \\ \tilde{A}_{ij} &= \phi_j(x_i) = \Phi(\|x_i - x_j\|), \quad i = N - N_B + 1, \dots, N, \quad j = 1, \dots, N. \end{aligned}$$

Assuming that the system matrix is invertible one then obtains the approximate solution (at any point x) by using the coefficients $\boldsymbol{\lambda}$ in (12). However, it is known that certain grids do not allow invertibility of the system matrix in (13) (see, e.g., the counterexamples in [10]). In [12] an iterative algorithm is suggested that adaptively builds grids for which the matrix is invertible.

4.2 An RBF-based Pseudospectral Method I

If we are interested in the solution at the grid points only, then (using $\boldsymbol{\lambda}$ from (13))

$$\mathbf{u} = A\boldsymbol{\lambda} = A \begin{bmatrix} \tilde{A}_{\mathcal{L}} \\ \tilde{A} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix}$$

with evaluation matrix A such that $A_{ij} = \phi_j(x_i)$ as above. This suggests that (according to our discussion in Section 2) the discretized differential operator L based on the grid points x_i , $i = 1, \dots, N$, and basis functions ϕ_j , $j = 1, \dots, N$, is given by

$$L_\Gamma = \begin{bmatrix} \tilde{A}_\mathcal{L} \\ \tilde{A} \end{bmatrix} A^{-1}.$$

Indeed, we have

$$\begin{bmatrix} \tilde{A}_\mathcal{L} \\ \tilde{A} \end{bmatrix} A^{-1} = \begin{bmatrix} M & P \\ 0 & I \end{bmatrix}$$

with the same blocks M , P , 0 and I as above. To see this we introduce the following notation:

$$A = \begin{bmatrix} \mathbf{a}_1^T \\ \vdots \\ \mathbf{a}_N^T \end{bmatrix} \quad \text{and} \quad A^{-1} = [\mathbf{a}_1^{-1} \quad \dots \quad \mathbf{a}_N^{-1}]$$

with column vectors \mathbf{a}_i and \mathbf{a}_j^{-1} such that $\mathbf{a}_i^T \mathbf{a}_j^{-1} = \delta_{ij}$. For Kansa's matrix from (13) this notation implies

$$\begin{bmatrix} \tilde{A}_\mathcal{L} \\ \tilde{A} \end{bmatrix} = \begin{bmatrix} \mathbf{a}_{\mathcal{L},1}^T \\ \vdots \\ \mathbf{a}_{\mathcal{L},N-N_B}^T \\ \mathbf{a}_{N-N_B+1}^T \\ \vdots \\ \mathbf{a}_N^T \end{bmatrix},$$

where we have used an analogous notation to denote the rows of the block $\tilde{A}_\mathcal{L}$. Now the discretized differential operator based on the non-symmetric collocation approach is given by

$$\begin{aligned} \begin{bmatrix} \tilde{A}_\mathcal{L} \\ \tilde{A} \end{bmatrix} A^{-1} &= \begin{bmatrix} \mathbf{a}_{\mathcal{L},1}^T \\ \vdots \\ \mathbf{a}_{\mathcal{L},N-N_B}^T \\ \mathbf{a}_{N-N_B+1}^T \\ \vdots \\ \mathbf{a}_N^T \end{bmatrix} [\mathbf{a}_1^{-1} \quad \dots \quad \mathbf{a}_N^{-1}] \\ &= \begin{bmatrix} \tilde{A}_\mathcal{L} A_I^{-1} & \tilde{A}_\mathcal{L} A_B^{-1} \\ \underbrace{\tilde{A} A_I^{-1}}_{=0} & \underbrace{\tilde{A} A_B^{-1}}_{=I} \end{bmatrix}. \end{aligned}$$

Here we partitioned A^{-1} into the blocks A_I^{-1} with $N - N_B$ columns corresponding to interior points, and A_B^{-1} with N_B columns corresponding to the remaining boundary points. Also, we made use of the fact that $\mathbf{a}_i^T \mathbf{a}_j^{-1} = \delta_{ij}$.

This is clearly the same as (see (10))

$$\begin{bmatrix} M & P \\ 0 & I \end{bmatrix} = L_\Gamma,$$

where M and P were obtained from the discrete differential operator

$$L = A_{\mathcal{L}}A^{-1} = \begin{bmatrix} A_{\mathcal{L}}A_I^{-1} & A_{\mathcal{L}}A_B^{-1} \end{bmatrix}$$

by replacing certain rows with unit vectors.

We have just seen that – provided we use the same basis functions ϕ_j and the same grid of collocation points x_i – the non-symmetric RBF collocation approach for the solution of an elliptic PDE with Dirichlet boundary conditions followed by evaluation at the grid points is identical to a pseudospectral approach. However, neither of the two methods is well-defined in general since they both rely on the invertibility of Kansa’s collocation matrix.

Above we showed that we can always form the discretized differential operator

$$L_{\Gamma} = \begin{bmatrix} \tilde{A}_{\mathcal{L}} \\ \tilde{A} \end{bmatrix} A^{-1} = \begin{bmatrix} M & P \\ 0 & I \end{bmatrix}$$

– even if Kansa’s matrix is not invertible. This implies that we can safely use the non-symmetric RBF pseudospectral approach whenever inversion of the discretized differential operator is not required (e.g., in the context of explicit time-stepping for parabolic PDEs).

Another interesting feature that we will illustrate below is the fact shown recently by a number of authors (see, e.g., [2], [4], [16]) that in the limiting case of “flat” basis functions (i.e., parameter $\varepsilon \rightarrow 0$ in (3–6)) the one-dimensional RBF interpolant yields a polynomial interpolant. Since we also mentioned earlier that the discretized differential operator L_{Γ} is invertible if a univariate polynomial basis is used we can conclude that Kansa’s collocation matrix is invertible in the limiting case $\varepsilon \rightarrow 0$.

4.3 A Symmetric Collocation Method

A second RBF collocation method has a symmetric system matrix that is known to be invertible for all grid configurations and any positive definite basic function (see, e.g., [5] and references therein). An interesting observation will be that Kansa’s matrix (actually its transpose) is the evaluation matrix for a PS approach based on the same set of basis functions used for this symmetric collocation approach.

For the symmetric collocation method one uses a different basis than in (12), i.e., a different function space than for the non-symmetric case. For the same elliptic PDE and boundary conditions as above one now starts with

$$u^h(x) = \sum_{j=1}^{N-N_B} \lambda_j \mathcal{L}_j^* \phi(x) + \sum_{j=N-N_B+1}^N \lambda_j \phi_j(x). \quad (14)$$

Since the ϕ_j are radial functions, i.e., $\phi_j(x) = \Phi(\|x - x_j\|)$ the functionals \mathcal{L}_j^* can be interpreted as an application of \mathcal{L} to Φ viewed as a function of the second variable followed by evaluation at x_j . One obtains the coefficients $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_{\Omega}, \boldsymbol{\lambda}_{\Gamma}]^T$ by solving the linear system

$$\begin{bmatrix} \hat{A}_{\mathcal{L}\mathcal{L}^*} & \hat{A}_{\mathcal{L}} \\ \hat{A}_{\mathcal{L}^*} & \hat{A} \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda}_{\Omega} \\ \boldsymbol{\lambda}_{\Gamma} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{g} \end{bmatrix}. \quad (15)$$

Here the blocks $\hat{A}_{\mathcal{L}\mathcal{L}^*}$ and \hat{A} , respectively, are square matrices corresponding to the interaction of interior grid points with each other and boundary grid points with each other. Their entries are given by

$$\begin{aligned}\hat{A}_{\mathcal{L}\mathcal{L}^*,ij} &= \left[\mathcal{L} [\mathcal{L}^* \Phi(\|x - \xi\|)]_{\xi=x_j} \right]_{x=x_i}, \quad i, j = 1, \dots, N - N_B, \\ \hat{A}_{ij} &= \phi_j(x_i) = \Phi(\|x_i - x_j\|), \quad i, j = N - N_B + 1, \dots, N.\end{aligned}$$

The other two blocks are rectangular, and correspond to interaction of interior points with boundary points and vice versa. They are defined as

$$\begin{aligned}\hat{A}_{\mathcal{L},ij} &= [\mathcal{L}\Phi(\|x - x_j\|)]_{x=x_i}, \quad i = 1, \dots, N - N_B, \quad j = N - N_B + 1, \dots, N, \\ \hat{A}_{\mathcal{L}^*,ij} &= [\mathcal{L}^*\Phi(\|x_i - x\|)]_{x=x_j}, \quad i = N - N_B + 1, \dots, N, \quad j = 1, \dots, N - N_B.\end{aligned}$$

As already mentioned, it is well known that the system matrix in (15) is invertible for positive definite radial functions. This implies that we can obtain the approximate solution at any point x by using the computed coefficients $\boldsymbol{\lambda}$ in (14). Thus this collocation method is very similar to Kansa's non-symmetric method with the notable difference that the collocation approach is well-defined.

4.4 An RBF-based Pseudospectral Method II

A nice connection between the symmetric and non-symmetric collocation methods appears if we consider the symmetric pseudospectral approach.

To this end we use the expansion (14) on which the symmetric collocation method is based as starting point for a pseudospectral method, i.e.,

$$u^h(x) = \sum_{j=1}^{N-N_B} \lambda_j \mathcal{L}_j^* \phi(x) + \sum_{j=N-N_B+1}^N \lambda_j \phi_j(x).$$

In vectorized notation this corresponds to

$$u^h(x) = \begin{bmatrix} \mathbf{a}_{\mathcal{L}^*}^T(x) & \tilde{\mathbf{a}}^T(x) \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda}_\Omega \\ \boldsymbol{\lambda}_\Gamma \end{bmatrix} \quad (16)$$

with appropriate row vectors $\mathbf{a}_{\mathcal{L}^*}^T(x)$ and $\tilde{\mathbf{a}}^T(x)$. Evaluated on the grid of collocation points this becomes

$$\mathbf{u} = \begin{bmatrix} A_{\mathcal{L}^*} & \tilde{A}^T \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda}_\Omega \\ \boldsymbol{\lambda}_\Gamma \end{bmatrix}.$$

Here the blocks $A_{\mathcal{L}^*}$ and \tilde{A}^T of the evaluation matrix are rectangular matrices with entries

$$\begin{aligned}A_{\mathcal{L}^*,ij} &= [\mathcal{L}^*\Phi(\|x_i - x\|)]_{x=x_j}, \quad i = 1, \dots, N, \quad j = 1, \dots, N - N_B, \\ \tilde{A}_{ij}^T &= \phi_j(x_i) = \Phi(\|x_i - x_j\|), \quad i = 1, \dots, N, \quad j = N - N_B + 1, \dots, N,\end{aligned}$$

corresponding to evaluation of the basis functions used in (14) at the grid points. Note that the second matrix with entries $\phi_j(x_i)$ is in fact the transpose of the corresponding

block of the system matrix in (13) for Kansa’s method (and thus use of the same notation is justified).

Moreover, the radial symmetry of the basic functions implies that the first block of the evaluation matrix for the symmetric collocation method is again the transpose of the corresponding block in Kansa’s collocation method. To see this we consider differential operators of even orders and odd orders separately. If \mathcal{L} is a linear differential operator of odd order, then \mathcal{L}^* will introduce a sign change (since it is acting on Φ as a function of the second variable). In addition, odd order derivatives (evaluated at $x = x_j$ include a factor of the form $x_i - x_j$. Now, transposition of this factor will again lead to a sign change. The combination of these two effects ensures that $A_{\mathcal{L}^*} = \tilde{A}_{\mathcal{L}}^T$. For even orders the effects of the operators \mathcal{L} and \mathcal{L}^* are indistinguishable.

Therefore, using symmetric RBF collocation we obtain the approximate solution of the boundary value problem on the grid as

$$\mathbf{u} = \begin{bmatrix} \tilde{A}_{\mathcal{L}} \\ \tilde{A} \end{bmatrix}^T \begin{bmatrix} \hat{A}_{\mathcal{L}\mathcal{L}^*} & \hat{A}_{\mathcal{L}} \\ \hat{A}_{\mathcal{L}^*} & \hat{A} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{f} \\ \mathbf{g} \end{bmatrix}.$$

We emphasize that this is not the solution of a pseudospectral method built on the same function space (same basis functions and same collocation points) as the symmetric RBF collocation method.

For a pseudospectral method we would require the discretized differential operator. Formally (assuming invertibility of Kansa’s matrix) we would have

$$\hat{L}_{\Gamma} = \begin{bmatrix} \hat{A}_{\mathcal{L}\mathcal{L}^*} & \hat{A}_{\mathcal{L}} \\ \hat{A}_{\mathcal{L}^*} & \hat{A} \end{bmatrix} \begin{bmatrix} \tilde{A}_{\mathcal{L}} \\ \tilde{A} \end{bmatrix}^{-T},$$

where we already incorporated the boundary conditions in a way analogous to our earlier discussion.

The problem with the second (symmetric) pseudospectral approach is that we cannot be assured that the method itself (i.e., the discretized differential operator) is well-defined. In fact, due to the Hon-Schaback counterexample [10] we know that there exist grid configurations for which the “basis” used for the symmetric PS expansion is not linearly independent.

Therefore, the symmetric RBF collocation approach is well-suited for problems that require inversion of the differential operator (such as elliptic PDEs). Subsequent evaluation on a grid makes the symmetric collocation look like a pseudospectral method – but it may not be (since we may not be able to formulate the pseudospectral *Ansatz*).

5 A Unified Discussion

In both the symmetric and non-symmetric collocation approaches we can think of the approximate solution as a linear combination of appropriate basis functions. In vectorized notation this can be written as

$$\mathbf{u}^h(x) = \mathbf{p}(x)\boldsymbol{\lambda}, \tag{17}$$

where the vector $\mathbf{p}(x)$ contains the basis functions at x . If we consider the non-symmetric method these basis functions are just ϕ_j , $j = 1, \dots, N$, while for the symmetric method they are comprised of both functions of the type ϕ_j and $\mathcal{L}_j^*\phi$ (cf. (16)).

We now let \mathcal{D} denote the linear operator that combines both the differential operator \mathcal{L} and the boundary operator (for Dirichlet boundary conditions this is just the identity). Then we have

$$\mathcal{D}u^h(x) = \mathcal{D}\mathbf{p}(x)\boldsymbol{\lambda} = \mathbf{q}(x)\boldsymbol{\lambda} \quad (18)$$

for an appropriately defined vector $\mathbf{q}(x)$. Therefore, the boundary value problem for our approximate solution is given by

$$\mathcal{D}u^h(x) = f(x),$$

where f is a piecewise defined function that collects the forcing functions in both the interior and on the boundary.

Now we evaluate the two representations (17) and (18) on the grid of collocation points x_i , $i = 1, \dots, N$, and obtain

$$\mathbf{u} = P\boldsymbol{\lambda} \quad \text{and} \quad \mathbf{u}_{\mathcal{D}} = Q\boldsymbol{\lambda}$$

with matrices P and Q whose rows correspond to evaluation of the vectors $\mathbf{p}(x)$ and $\mathbf{q}(x)$, respectively, at the collocation points x_i . The discretized boundary value problem is then

$$\mathbf{u}_{\mathcal{D}} = Q\boldsymbol{\lambda} = \mathbf{f}, \quad (19)$$

where \mathbf{f} is the vector of values of f on the grid.

For the non-symmetric collocation approach P is the standard RBF interpolation matrix, and Q is Kansa's matrix, whereas for symmetric collocation P is the transpose of Kansa's matrix, and Q is the symmetric collocation matrix.

It is our goal to find the vector \mathbf{u} , i.e., the values of the approximate solution on the grid. There are two ways by which we can obtain this answer:

1. We solve $Q\boldsymbol{\lambda} = \mathbf{f}$ for $\boldsymbol{\lambda}$, i.e.,

$$\boldsymbol{\lambda} = Q^{-1}\mathbf{f}.$$

Then we use the discretized version of (17) to get the desired vector \mathbf{u} as

$$\mathbf{u} = PQ^{-1}\mathbf{f}.$$

2. Alternatively, we first transform the coefficients, i.e., we rewrite $\mathbf{u} = P\boldsymbol{\lambda}$ as

$$\boldsymbol{\lambda} = P^{-1}\mathbf{u}.$$

Then the discretized boundary value problem (19) becomes

$$QP^{-1}\mathbf{u} = \mathbf{f},$$

and we can obtain the solution vector \mathbf{u} by solving this system.

The first approach corresponds to RBF collocation, the second to the pseudospectral approach. Both of these approaches are equivalent as long as all of the matrices involved are invertible. Unfortunately, as mentioned earlier, there are configurations of grid points for which Kansa's matrix is not invertible. This means that for the non-symmetric case (Q is Kansa's matrix) Approach 1 cannot be assured to work in general, and Approach 2 can only be used if the discretized differential operator is applied directly (but not inverted). For the symmetric approach (P is Kansa), on the other hand, Approach 1 is guaranteed to work in general, but Approach 2 may not be well-defined.

6 Numerical Experiments

In this section we illustrate how the RBF pseudospectral approach can be applied in a way completely analogous to standard polynomial pseudospectral methods. Among our numerical illustrations are two examples taken directly from the book [17] by Trefethen (see Programs 35 and 36 there). We will also use a 1-D transport equation to compare the RBF and polynomial pseudospectral methods. Plots of the spectra of the respective differentiation matrices will support the theoretical results mentioned earlier which state that RBF methods tend to polynomial methods in the limit $\varepsilon \rightarrow 0$.

We begin by giving a brief explanation of how the Contour-Padé algorithm of Fornberg and Wright [7] can be used to compute the discretized differential operators (differentiation matrices) mentioned in our earlier discussions.

6.1 Use of the Contour-Padé Algorithm with the PS Approach

In its original form the Contour-Padé algorithm of Fornberg and Wright [7] allows one to stably evaluate radial basis function interpolants based on infinitely smooth RBFs for extreme choices of the shape parameter ε (in particular $\varepsilon \rightarrow 0$). More specifically, the Contour-Padé algorithm uses FFTs and Padé approximations to evaluate the function

$$s(x, \varepsilon) = \mathbf{p}(\varepsilon)(A(\varepsilon))^{-1} \mathbf{f} \quad (20)$$

with $\mathbf{p}(\varepsilon)_j = \Phi(\|x - x_j\|)$ and $A(\varepsilon)_{j,k} = \Phi(\|x_k - x_j\|)$ at some evaluation point x (cf. the discussion in the previous section). The parameter ε is used to denote the dependence of \mathbf{p} and A on the choice of that parameter in the basic function Φ .

If we evaluate s at all of the grid points x_j for some fixed value of ε , then $\mathbf{p}(\varepsilon)$ turns into the matrix $A(\varepsilon)$. In the case of interpolation this exercise is, of course, pointless. However, if the Contour-Padé algorithm is adapted to replace $\mathbf{p}(\varepsilon)$ with the matrix $A_{\mathcal{L}}$ based on the differential operator used earlier, then

$$\mathbf{p}(\varepsilon)(A(\varepsilon))^{-1} \mathbf{u} = A_{\mathcal{L}}(\varepsilon)(A(\varepsilon))^{-1} \mathbf{u}$$

computes the values of the (spatial) derivative of \mathbf{u} on the grid points x_j . Boundary conditions can then be incorporated later as in the standard pseudospectral approach (see, e.g., [17] or our discussion in Section 3).

This means that we can take standard pseudospectral code (such as that presented in [17]) and replace just one subroutine, namely that which provides the differentiation matrix, by the appropriate one for radial basis functions. In the examples given below we will use both Contour-Padé and straightforward (based on LU-factorization of the collocation systems) implementations of these subroutines.

6.2 Example: 1-D Transport Equation

Consider

$$\begin{aligned} u_t(x, t) + cu_x(x, t) &= 0, & x > -1, t > 0, \\ u(-1, t) &= 0, \\ u(x, 0) &= f(x), \end{aligned}$$

with solution

$$u(x, t) = f(x - ct).$$

The graph in Figure 1 shows the time profile of the solution for the time interval $[0, 1]$ with initial profile $f(x) = 64(1 - x)^3x^3$ and unit wave speed. For the computation we used Gaussian RBFs (4) and the Contour-Padé algorithm in the limiting case $\varepsilon \rightarrow 0$ together with an implicit Euler method with time step $\Delta t = 0.01$ for the time discretization. We point out that this, of course, requires an inversion of the differentiation matrix. Recall that our earlier theoretical discussion suggested that this is possible as long as we limit ourselves to the limiting case $\varepsilon \rightarrow 0$ and one space dimension. Also, an implementation based on an explicit Euler method showed almost identical behavior for a smaller time step so that we can be assured that the inversion was indeed justified for this particular example. The explicit Euler method, on the other hand, exhibited instabilities for the larger time step used in Figure 1. The exact stability conditions (CFL conditions) for the RBF pseudospectral method are an open problem. The spatial discretization for the solution displayed in Figure 1 was based on 19 Chebyshev points in $[-1, 1]$.

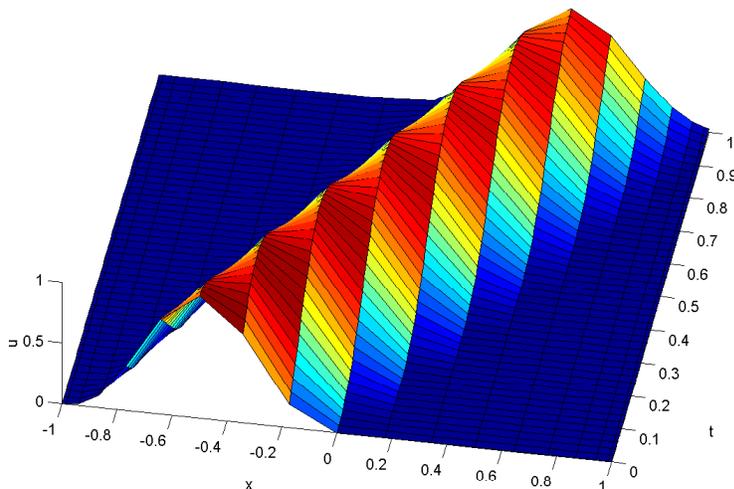


Figure 1: Solution to transport equation based on Gaussian RBFs with Contour-Padé ($\varepsilon = 0$), implicit Euler ($\Delta t = 0.01$), and 19 Chebyshev points.

In Figure 2 we plot the maximum errors at time $t = 1$ for two different time steps ($\Delta t = 0.01$ and $\Delta t = 0.001$) and spatial discretizations consisting of 5 up to 19 points. If we use Chebyshev polynomials instead of Gaussian RBFs then the error plots look identical (and are therefore omitted).

The spectra of the differentiation matrices for both the Gaussian and the Chebyshev PS approaches are plotted in Figures 3 and 4, respectively. The subplots correspond to the use of $N = 5, 9, 13, 17$ Chebyshev points for the spatial discretization. The plots for the Gaussian and Chebyshev methods are almost identical. There is only a slight difference in the location of the eigenvalues in the case $N = 17$. We point out that one

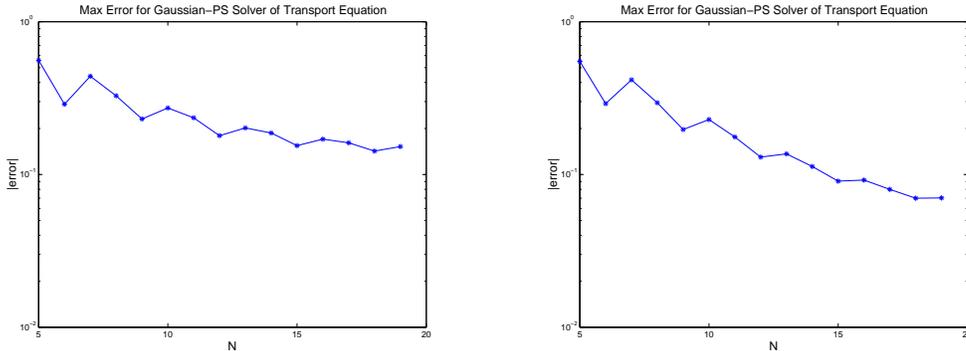


Figure 2: Errors at $t = 1$ for transport equation. Gaussian RBFs ($\varepsilon = 0$), variable spatial discretization N . Implicit Euler method with $\Delta t = 0.01$ (left), $\Delta t = 0.001$ (right).

of the (severe) limitations of the Contour-Padé algorithm is its guaranteed accuracy for very small problems only. Thus, already for the case $N = 17$ this feature is slightly visible.

6.3 Example: Allen-Cahn Equation

Next, we illustrate the solution of a nonlinear reaction-diffusion equation. To be specific we adapt Program 35 of [17] involving the nonlinear Allen-Cahn equation

$$u_t = \mu u_{xx} + u - u^3, \quad x \in (-1, 1), \quad t \geq 0,$$

with parameter μ , initial condition

$$u(x, 0) = 0.53x + 0.47 \sin\left(-\frac{3}{2}\pi x\right), \quad x \in [-1, 1],$$

and non-homogeneous (time-dependent) boundary conditions $u(-1, t) = -1$ and $u(1, t) = \sin^2(t/5)$. This equation has three steady solutions ($u = -1, 0, 1$) with the two nonzero solutions being stable. The transition between these states is governed by the parameter μ . In our calculations below we use $\mu = 0.01$, and the unstable state should vanish around $t = 30$. Sample Matlab code to solve this problem using an explicit Euler discretization for the time-derivative and a Chebyshev pseudospectral differentiation matrix for the spatial derivative is listed in Table 1 (page 141 of [17]).

Essentially, one needs only to form the differentiation matrix for the second spatial derivative (which can be taken as the square of the first derivative matrix, i.e., $D^2 = D^2$, see line 2 of the algorithm) and use this within the time-stepping method that incorporates the nonlinearity of the problem (see line 11).

We can apply the code from [17] almost verbatim for radial basis functions. In [17] the differentiation matrix is obtained by a call to the subroutine `cheb.m` in line 2

```

% p35.m - Allen-Cahn eq. as in p34.m, but with boundary condition
%           imposed explicitly ("method (II)")

% Differentiation matrix and initial data:
1  figure(1);
2  N = 20; [D,x] = cheb(N); D2 = D^2;
3  eps = 0.01; dt = min([.01,50*N^(-4)/eps]);
4  t = 0; v = .53*x + .47*sin(-1.5*pi*x);

% Solve PDE by Euler formula and plot results:
5  tmax = 100; tplot = 2; nplots = round(tmax/tplot);
6  plotgap = round(tplot/dt); dt = tplot/plotgap;
7  xx = -1:.025:1; vv = polyval(polyfit(x,v,N),xx);
8  plotdata = [vv; zeros(nplots,length(xx))]; tdata = t;
9  for i = 1:nplots
10     for n = 1:plotgap
11         t = t+dt; v = v + dt*(eps*D2*v + v - v.^3);           % Euler
12         v(1) = 1 + sin(t/5)^2; v(end) = -1;                   % BC
13     end
14     vv = polyval(polyfit(x,v,N),xx);
15     plotdata(i+1,:) = vv; tdata = [tdata; t];
16 end
17 clf, subplot('position',[.1 .4 .8 .5])
18 mesh(xx,tdata,plotdata), grid on, axis([-1 1 0 tmax -1 2]),
19 view(-60,55), colormap(1e-6*[1 1 1]); xlabel x, ylabel t, zlabel u

```

Table 1: Program 35 of [17]

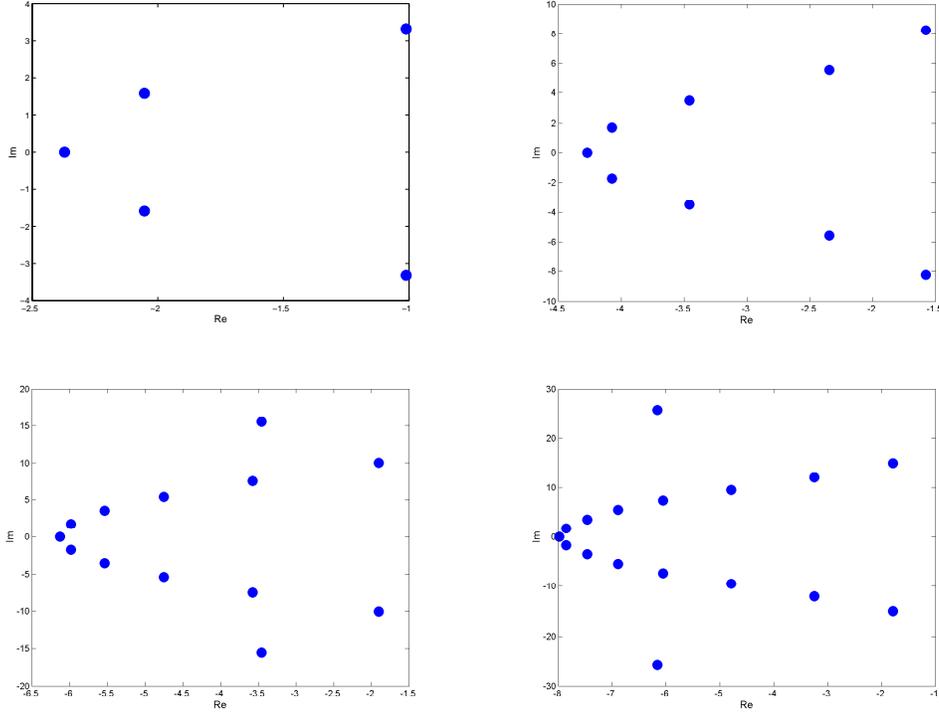


Figure 3: Spectra of differentiation matrices for Gaussian RBF ($\varepsilon = 0$). $N = 5, 9, 13, 17$ Chebyshev points.

which yields the matrix D for the discretization of the first derivative operation on the Chebyshev points. The only difference is to replace this by a call to a subroutine `DRBF.m` that generates the RBF differentiation matrix $D = A_{\mathcal{L}}A^{-1}$ as explained earlier (see (9)). Thus, we replace line 2 of the algorithm with

```
2' N = 20; [D,x] = DRBF(N); D2 = D^2;
```

or by using second derivatives of the RBFs to generate the matrix A_L , i.e.,

```
2' N = 20; [D2,x] = D2RBF(N);
```

With the latter approach `D2` will be a discretization of the second spatial derivative on the grid points (which may be arbitrarily spaced for the use with RBFs).

The implementation of `DRBF.m` is accomplished either via the Contour-Padé algorithm of Fornberg and Wright as explained above, or by explicitly setting up a first derivative matrix $A_{\mathcal{L}}$ and an evaluation matrix (i.e., interpolation matrix) A and computing its inverse. Note that the majority of the matrix computations are required only once outside the time-stepping procedure. Inside the time-stepping loop (lines 10–13) we only require matrix-vector multiplication. We point out that this approach is much

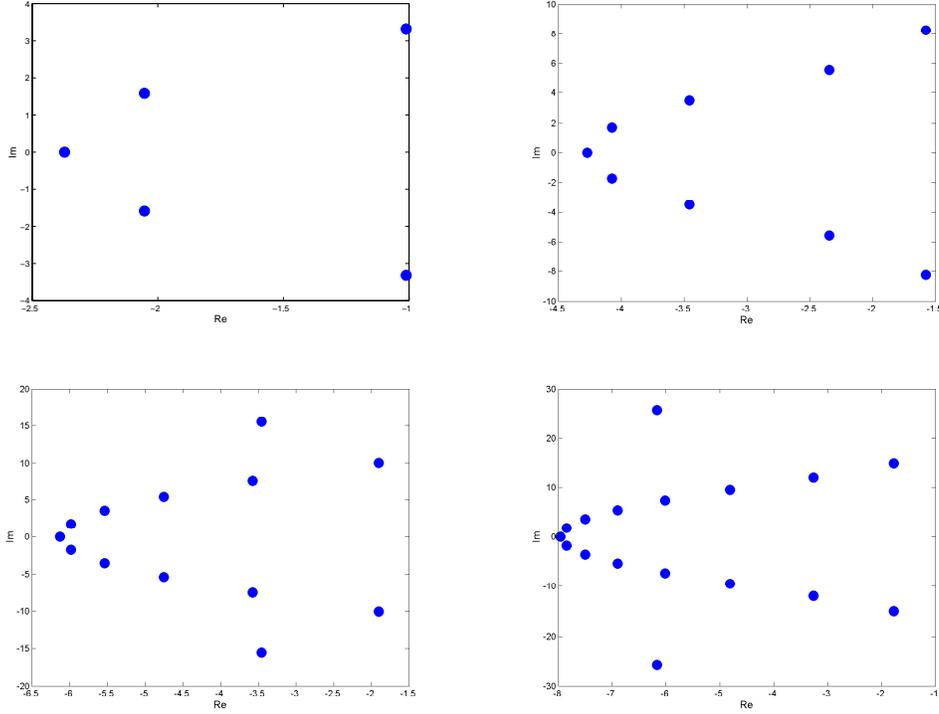


Figure 4: Spectra of differentiation matrices for Chebyshev pseudospectral method. $N = 5, 9, 13, 17$ Chebyshev points.

more efficient than computation of RBF expansion coefficients at every time step (as suggested, e.g., in [9]).

Figures 5 and 6, respectively, show the solution obtained with the Gaussian RBF (with $\varepsilon = 0$ via Contour-Padé) and with the Chebyshev pseudospectral method. Both spatial discretizations are based on $N = 10$ Chebyshev points in $[-1, 1]$. Clearly, a spatial discretization of only 10 points does not suffice. The metastable solution only “survives” for t up to about $t = 20$ (instead of $t \approx 30$ for an accurate solution). However, again, the RBF and polynomial PS solutions are identical.

In Figures 7 and 8, respectively, we show the solution obtained via the Chebyshev pseudospectral method and via an RBF pseudospectral approach based on the Matérn function (5) with $\varepsilon = 2.0$. These computations were based on 20 Chebyshev points, and the differentiation matrix for the RBF was obtained directly (i.e., without the Contour-Padé algorithm). We used this approach since for 20 points the Contour-Padé algorithm no longer can be relied upon. Moreover, it is apparent from the figures that reasonable solutions can also be obtained via this direct (and much simpler) RBF approach. Spectral accuracy, however, will no longer be given if $\varepsilon > 0$.

The solution based on Chebyshev polynomials for $N = 20$ is the most accurate since the transition occurs at the correct time (i.e., at $t \approx 30$) and is a little “sharper”.

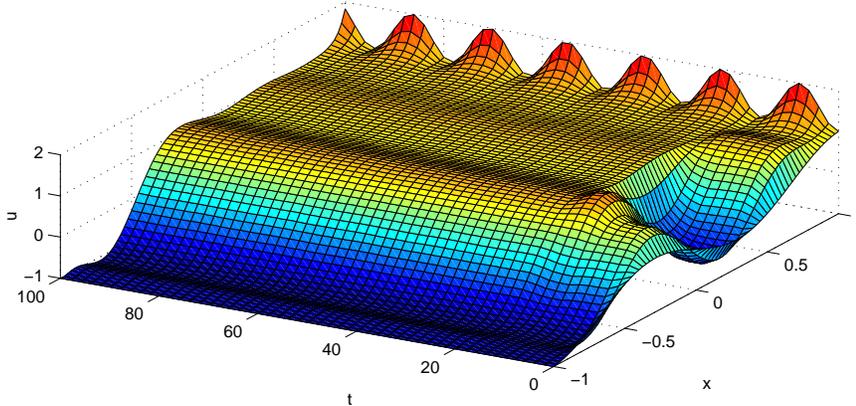


Figure 5: Solution of the Allen-Cahn equation using Gaussian RBFs ($\varepsilon = 0$, $N = 10$).

6.4 Example: 2-D Laplace Equation

Our final example is an elliptic equation. For such a problem inversion of the differentiation matrix is required. Even though this may not be warranted theoretically, we compare an RBF pseudospectral method based on the non-symmetric Kansa *Ansatz* with a Chebyshev pseudospectral method. As in the previous example, the Matérn RBF (5) is used (this time with $\varepsilon = 2.4$), and the (inverse of the) differentiation matrix is computed using standard Matlab routines.

We consider the 2-D Laplace equation

$$u_{xx} + u_{yy} = 0, \quad x, y \in (-1, 1)^2,$$

with boundary conditions

$$u(x, y) = \begin{cases} \sin^4(\pi x), & y = 1 \text{ and } -1 < x < 0, \\ \frac{1}{5} \sin(3\pi y), & x = 1, \\ 0, & \text{otherwise.} \end{cases}$$

This is the same problem as used in Program 36 of [17].

Figures 9 and 10 show the solution obtained via the RBF and Chebyshev pseudospectral methods, respectively. Now the spatial discretization consists of a tensor product of $N = 24 \times 24$ Chebyshev points.

The qualitative behavior of the two solutions is very similar. While there is no advantage in going to arbitrarily irregular grid points for any of the problems presented here, there is nothing that prevents us from doing so for the RBF approach. In particular, we are not limited to using tensor product grids for higher-dimensional spatial discretizations. This is a potential advantage of the RBF pseudospectral approach over the standard polynomial methods.

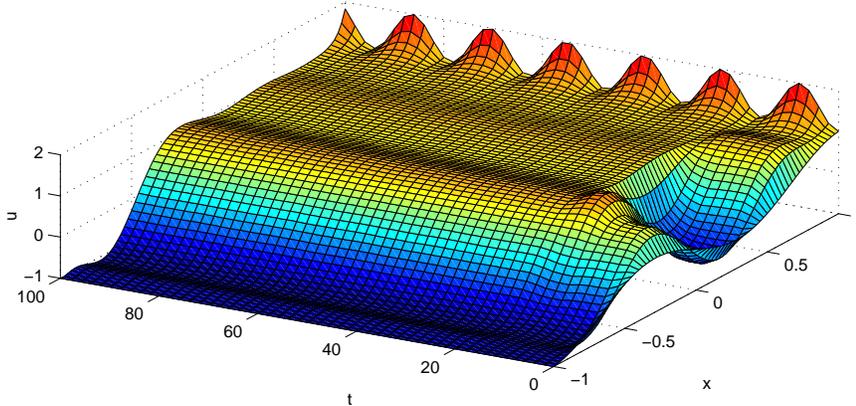


Figure 6: Solution of the Allen-Cahn equation using the Chebyshev pseudospectral method ($N = 10$).

7 Summary

In this paper we attempted to establish a connection between RBF collocation methods and standard (polynomial) pseudospectral methods. Our discussion revealed that for the non-symmetric (Kansa) *Ansatz* (12) we can always formulate the discrete differential operator

$$L_{\Gamma} = \begin{bmatrix} \tilde{A}_{\mathcal{L}} \\ \tilde{A} \end{bmatrix} A^{-1}.$$

However, we cannot ensure in general the invertibility of L_{Γ} . This implies that the non-symmetric RBF pseudospectral approach is justified for time-dependent PDEs (with explicit time-stepping methods).

For the symmetric *Ansatz* (14), on the other hand, we can in general ensure the solution of $\mathcal{L}u = f$. However, it is not possible in general to even formulate the discrete differential operator

$$\hat{L}_{\Gamma} = \begin{bmatrix} \hat{A}_{\mathcal{L}\mathcal{L}^*} & \hat{A}_{\mathcal{L}} \\ \hat{A}_{\mathcal{L}^*} & \hat{A} \end{bmatrix} \begin{bmatrix} A_{\mathcal{L}^*} & \tilde{A}^T \end{bmatrix}^{-1}.$$

This suggests that we should use the symmetric approach for time-dependent PDEs with implicit time-stepping as well as for time-independent PDEs.

The difficulties with both approaches can be attributed to the possible singularity of Kansa's matrix which appears as discretized differential operator for the non-symmetric approach, and (via its transpose) as the evaluation matrix in the symmetric approach.

Since the non-symmetric approach is quite a bit easier to implement than the symmetric approach, and since the grid configurations for which the Kansa matrix is singular seem to be very rare (see [10]) many researchers (include ourselves) often prefer to

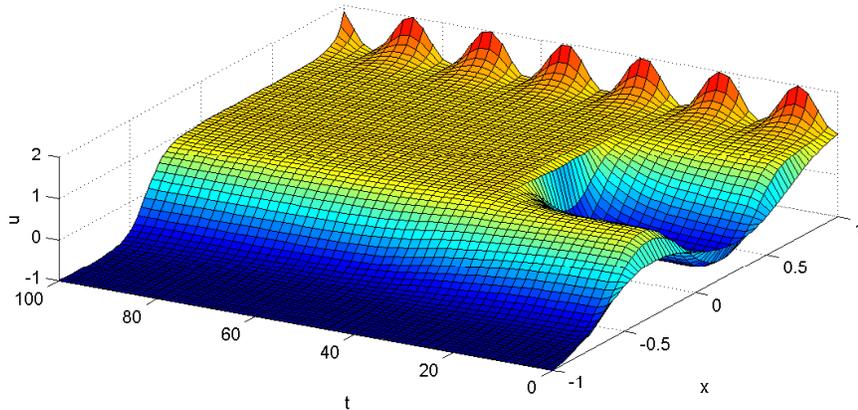


Figure 7: Solution of the Allen-Cahn equation using the Chebyshev pseudospectral method ($N = 20$).

use the non-symmetric approach – even under questionable circumstances (such as with implicit time-stepping procedures, or for elliptic problems). However, the connection to polynomials in the limiting case $\varepsilon = 0$ justifies this at least for 1-D problems.

Overall, the coupling of RBF collocation and pseudospectral methods obtained here has provided a number of new insights. For example, it should now be clear that we can apply many standard pseudospectral procedures to RBF solvers. However, with RBF expansions we can also take advantage of scattered (multivariate) grids as well as spatial domains with non-rectangular geometries. Thus, we now have “standard” procedures for solving time-dependent PDEs with RBFs. Moreover, we have illustrated that RBF pseudospectral methods for $\varepsilon = 0$ are identical to Chebyshev pseudospectral methods.

Future challenges include the problem of dealing with larger problems in an efficient and stable way. Thus, such issues as preconditioning and FFT-type algorithms need to be studied in the context of RBF pseudospectral methods. Some first results in this directions have been reported very recently in [14].

Another possible avenue opened up by the use of RBFs instead of polynomials is the study of pseudospectral methods with moving (adaptive) grids. This will be computationally much more involved, but the use of RBFs should imply that there is no major restriction imposed by moving (scattered) grids.

Acknowledgement: I would like to thank Robert Schaback for discussing with me at length the connection between the pseudospectral approach and RBF collocation, and Grady Wright for providing me with a pre-beta version of his Matlab toolbox for the Contour-Padé algorithm of [7].

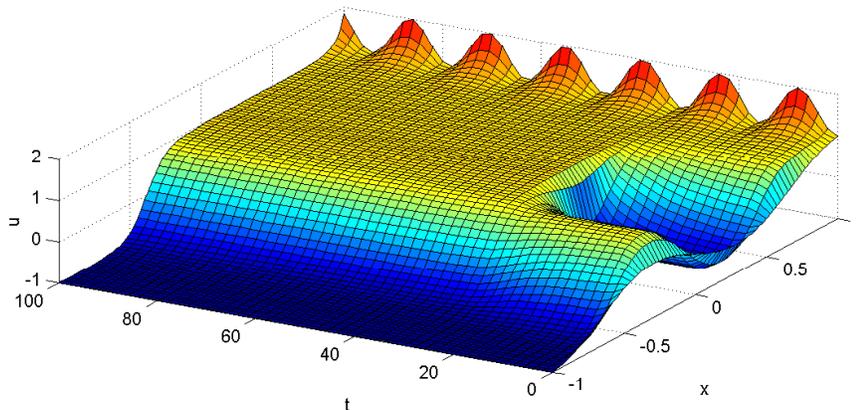


Figure 8: Solution of the Allen-Cahn equation using the Matérn RBF (5) ($N = 20$).

References

- [1] R. K. Beatson and H.-Q. Bui, Mollification formulas and implicit smoothing, Research Report UCDMS 2003/19, University of Canterbury, 2003.
- [2] C. de Boor, On interpolation by radial polynomials, *Adv. in Comput. Math.*, to appear.
- [3] C. Camuto, M. Y. Hussaini, A. Quarteroni, and T. A. Zang, *Spectral Methods in Fluid Dynamics*, Springer Verlag, Berlin, 1988.
- [4] T. A. Driscoll and B. Fornberg, Interpolation in the limit of increasingly flat radial basis functions, *Comput. Math. Appl.* 43 (2002), 413–422.
- [5] G. E. Fasshauer, Solving partial differential equations by collocation with radial basis functions, in *Surface Fitting and Multiresolution Methods*, A. Le Méhauté, C. Rabut, and L. L. Schumaker (eds.), Vanderbilt University Press, Nashville TN, 1997, 131–138.
- [6] B. Fornberg, *A Practical Guide to Pseudospectral Methods*, Cambridge Univ. Press, 1998.
- [7] B. Fornberg and G. Wright, Stable computation of multiquadric interpolants for all values of the shape parameter, *Comput. Math. Appl.* 47 (2004), 497–523.
- [8] D. Gottlieb and L. Lustman, The spectrum of the Chebyshev collocation operator for the heat equation, *SIAM J. Numer. Anal.* 20 (1983), no. 5, 909–921.
- [9] Y. C. Hon and X. Z. Mao, A radial basis function method for solving options pricing model, *Financial Engineering* (8) 1999, 31–49.

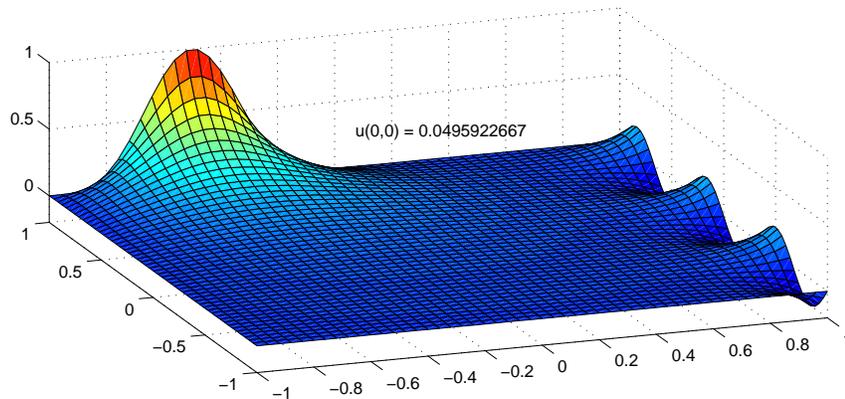


Figure 9: Solution of the Laplace equation using the Matérn RBF (5) ($\varepsilon = 2.4$, $N = 24 \times 24$).

- [10] Y. C. Hon and R. Schaback, On nonsymmetric collocation by radial basis functions, *Appl. Math. Comput.* 119 (2001), 177–186.
- [11] E. J. Kansa, 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 (1990), 147–161.
- [12] L. Ling, R. Opfer, and R. Schaback, Results on meshless collocation techniques, preprint, Universität Göttingen, 2004.
- [13] R. B. Platte and T. A. Driscoll, Polynomials and potential theory for Gaussian radial basis function interpolation, Technical report, University of Delaware, 2004.
- [14] R. B. Platte and T. A. Driscoll, Stabilizing radial basis function methods for time-dependent problems, *Comput. Math. Appl.*, submitted.
- [15] R. Schaback, On the efficiency of interpolation by radial basis functions, in *Surface Fitting and Multiresolution Methods*, A. Le Méhauté, C. Rabut, and L. L. Schumaker (eds.), Vanderbilt University Press, Nashville TN, 1997, 309–318.
- [16] R. Schaback, Multivariate interpolation by polynomials and radial basis functions, preprint, Universität Göttingen, 2002.
- [17] L. N. Trefethen, *Spectral Methods in Matlab*, SIAM, Philadelphia, PA, 2000.
- [18] H. Wendland, *Scattered Data Approximation*, Cambridge University Press, Cambridge, to appear.

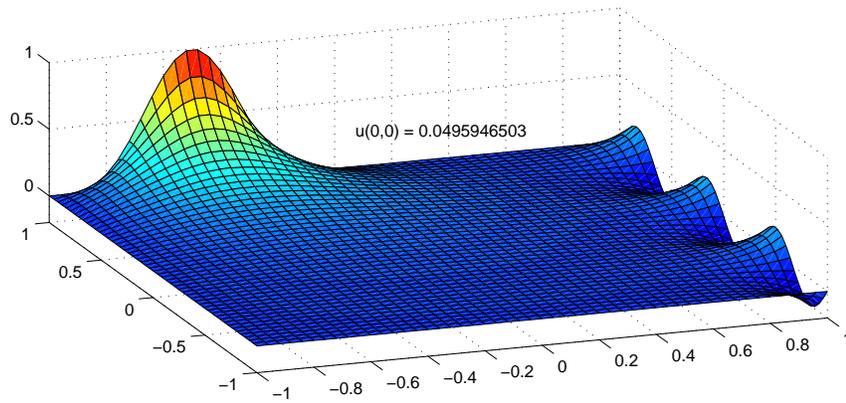


Figure 10: Solution of the Laplace equation using the Chebyshev pseudospectral method ($N = 24 \times 24$).