

Greedy Unsymmetric Collocation

Prasanth B. Nair⁽¹⁾

Abstract: *We present greedy unsymmetric collocation schemes for solving linear elliptic partial differential equations using radial basis functions. The proposed approach circumvents the ill-conditioning problem associated with the standard collocation technique and enables the efficient solution of problems requiring a large set of collocation points. Numerical studies indicate that the accuracy of greedy algorithms incorporating shape parameter tuning can be significantly better than the standard collocation scheme.*

Keywords: Radial basis functions, collocation, greedy algorithms.

1 Introduction

In recent years, much research has focused on the development of meshfree algorithms based on radial basis functions (RBFs) to solve partial differential equations (PDEs); see, for example, [1, 2, 3, 4]. To illustrate, consider a linear elliptic PDE of the form

$$Lu(x) = f(x) \text{ in } \Omega \subset \mathbb{R}^d, \quad \text{subject to: } Bu(x) = g(x) \text{ in } \partial\Omega, \quad (1)$$

where L and B are differential operators in space $x \in \mathbb{R}^d$, and $u(x)$ denotes the field variable. Ω is a bounded computational domain and $\partial\Omega$ denotes its boundary. In meshfree schemes based on RBFs, the field variable is approximated as

$$u(x) \approx \hat{u}(x) = \text{span} \{ \phi(\|x - x_1\|), \phi(\|x - x_2\|), \dots, \phi(\|x - x_n\|) \} = \sum_{i=1}^n \alpha_i \phi_i(x), \quad (2)$$

where $\phi_i \equiv \phi(\|x - x_i\|) : \mathbb{R}^d \rightarrow \mathbb{R}$ is a RBF centered at x_i and $\alpha_i, i = 1, 2, \dots, n$ are undetermined coefficients.

The vector of undetermined coefficients can be computed by Kansa's unsymmetric collocation scheme [1, 2], which involves first defining a set of nodes situated within the domain as well as on the boundary, i.e., $C = \{x_i |_{i=1, n_d} \subset \Omega, x_i |_{i=n_d+1, n_d+n_b} \subset \partial\Omega\}$, where $n = n_d + n_b$ is the total number of collocation points. In practice, the centers of the RBFs are chosen to coincide with the points in the set C and the undetermined coefficient vector α is calculated by collocating the governing equations on the set C . This step results

¹Computational Engineering and Design Group, University of Southampton, Highfield, Southampton SO17 1BJ, Email:P.B.Nair@soton.ac.uk, WWW: <http://www.soton.ac.uk/~pbn>

in a linear algebraic system of equations of the form $A\alpha = y$, where $A \in \mathbb{R}^{n \times n}$ is the unsymmetric collocation matrix. In practice, A can be highly ill-conditioned, particularly when increasingly flat globally supported RBFs are used.

In this paper, we propose greedy collocation schemes for solving PDEs which aim to: (i) avoid the ill-conditioning problem associated with the unsymmetric collocation scheme, (ii) reduce computational cost and memory requirements, and (iii) enable efficient tuning of the shape parameter of each RBF in Eqn. (2) to ensure good accuracy. We propose to achieve these goals by adaptively choosing ϕ_i in an iterative fashion by monitoring the spatial distribution of the residual error in Ω and $\partial\Omega$. To illustrate, let $\mathcal{D} := \{\phi_1, \phi_2, \dots, \phi_n\}$ denote a *dictionary* of RBFs and suppose that a baseline approximation for the field variable is given, say $u^k(x) = \sum_{i=1}^k \alpha_i \phi_i$, where $\phi_i \in \mathcal{D}$. Then the spatial distribution of the residual error in the domain and on the boundary can be written as $r_d(x) = Lu^k(x) - f$ and $r_b(x) = Bu^k(x) - g$, respectively. The basic idea of the proposed stage-wise adaptive scheme is to greedily choose a new RBF ϕ_{k+1} from the dictionary \mathcal{D} so as to minimize a suitable norm of r_d and r_b . Subsequently, the baseline model is updated as $u^{k+1}(x) \leftarrow u^k(x) + \alpha_{k+1} \phi_{k+1}(x)$. The iterations are terminated when $\|r_d(x)\|$ and $\|r_b(x)\|$ are smaller than a specified upper bound.

The idea of function approximation using greedy algorithms is not new – a number of papers devoted to this topic can be found in the signal processing, machine learning and numerical linear algebra literature; see, for example, [5, 7, 8]. It is of interest to note that Hon et. al. [6] proposed a greedy algorithm for solving PDEs. Even though encouraging results were obtained for a Poisson problem, it was observed that the greedy algorithm suffered from slow convergence, thereby requiring a large set of basis functions. Note that in contrast to the greedy algorithm in [6], the proposed schemes also implicitly perform a back-fitting procedure to update the coefficients of the RBFs chosen at previous iterations. This in turn results in a much faster rate of convergence. We present some numerical studies on a Poisson problem to illustrate some features of the proposed approach.

2 Greedy QR Collocation

The standard collocation procedure based on globally supported RBFs requires $O(n^2)$ memory and $O(n^3)$ operations when a direct method is used to compute α . One way to reduce these requirements would be to choose a subset of RBFs (say m basis functions) to approximate $u(x)$. This leads to the overdetermined least-squares problem: $\min \|\tilde{A}\tilde{\alpha} - y\|$, where $\tilde{A} \in \mathbb{R}^{n \times m}$ and $\tilde{\alpha} \in \mathbb{R}^m$. As a consequence, the memory requirements will reduce to $O(mn)$ and the number of operations required to compute α will also be reduced. The aim here is to (sub)optimally choose a subset of the RBFs to circumvent ill-conditioning while ensuring high accuracy. The problem of choosing a good set of basis functions from a dictionary for the model PDE in Eqn. (1) can be stated as follows:

Problem (P1): *Given a dictionary of RBFs $\mathcal{D} := \{\phi_1, \phi_2, \dots, \phi_n\}$, and $\varepsilon > 0$, find the smallest subset of m functions $\phi_i, i = 1, 2, \dots, m$ from \mathcal{D} and constants $\alpha_i, i = 1, 2, \dots, m$, if it exists, such that $\|L \sum_{i=1}^m \alpha_i \phi_i - f\| \leq \varepsilon$ and $\|B \sum_{i=1}^m \alpha_i \phi_i - g\| \leq \varepsilon$.*

It is of interest to note that (P1) is related to the *minimum set-cover problem* in theoretical computer science [5] and hence it can be shown that (P1) is NP-hard. This motivates the development of greedy algorithms to efficiently compute a suboptimal solution.

The template of a greedy algorithm to solve (P1) is shown in Algorithm 1. Here, I^p denotes the set of cardinality p which contains the indices of the RBFs chosen from the

Algorithm 1: Template of a greedy collocation scheme

Inputs: A dictionary of RBFs $\mathcal{D} := \{\phi_1, \phi_2, \dots, \phi_n\}$
and tolerance for residual ε .

Set $p = 0$, $\alpha^0 = 0$ and $I^p = []$

while $\|r^p\| < \varepsilon$, **do**

1. $p \leftarrow p + 1$.
2. Find $i_p = \arg \max_{j \notin I^{p-1}} J_j$
3. $I^p \leftarrow [I^{p-1}, i_p]$
4. Compute $A_{i_p} = \{L\phi_{i_p}(x_1), L\phi_{i_p}(x_2), \dots, L\phi_{i_p}(x_{n_d}),$
 $B\phi_{i_p}(x_{n_d+1}), B\phi_{i_p}(x_{n_d+2}), \dots, B\phi_{i_p}(x_n)\}^T \in \mathbb{R}^n$
5. Update α
6. Update r^p

end

dictionary \mathcal{D} at iteration p . A_{i_p} denotes the i_p th column of the collocation matrix A . It can be noted that the core of the algorithm consists of two steps – (i) selection of an index i_p (or RBF) at each iteration (in Step 2) by finding the basis function in the dictionary that leads to a maximum value of the criterion J and (ii) updating the weights α and the residual r to reflect the fact that a new RBF has been appended to the approximation for the field variable $u(x)$.

A number of criteria J can be used to greedily select a RBF from \mathcal{D} at each iteration; see Nair et al. [8] for an overview. For example, one simple and cheap way to compute i_p is to search for that point in the set \mathcal{C} where the residual error is highest. This point can then be used as a center for the new RBF. Once a new RBF has been appended to the approximation, the weight vector α and the residual r can be updated using gradient descent techniques such as those presented in [6, 7, 8]. In practice, however, it is preferable to use an incremental QR factorization scheme to update α and r since numerical studies on regression problems have shown that it tends to converge much faster than its counterparts [8]. Further, the incremental QR factorization scheme allows us to efficiently monitor the condition number of the collocation matrix when more RBFs are appended to the approximation. As a consequence, the numerical ill-conditioning problem associated with the standard collocation scheme can be circumvented. In addition, the incremental QR factorization scheme only incurs $O(np)$ operations and memory at iteration p . It is worth noting that the memory requirements of the greedy algorithm is low since only one column of the collocation matrix is computed at each iteration.

3 Local Shape Parameter Tuning

It is well known that by tuning the shape parameter of each RBF, a higher level of accuracy can be obtained by using a smaller number of terms. Heuristic recipes are often used in the literature for optimizing the shape parameter since approaches based on non-linear optimization tend to be computationally expensive. It so turns out that the greedy approach outlined in Algorithm 1 can be readily modified to include an additional step where the shape parameter σ is tuned for each RBF selected from the dictionary \mathcal{D} . A straight forward way to achieve this would be to minimize a suitable norm of the residual

error using a one-dimensional minimization technique. The residual error can be readily computed as an implicit function of the shape parameter when an incremental QR factorization scheme is used for updating α and r . However, for computational efficiency, it may be more preferable to solve the following one-dimensional minimization problem for each RBF i_p selected in Step 2 of Algorithm 1:

$$\sigma_{i_p} = \operatorname{argmin}_{\sigma} \|r^{p-1} + \mu_{i_p} A_{i_p}\|_2, \quad \text{where } \mu_{i_p} = -\frac{(r^{p-1}, A_{i_p})}{(A_{i_p}, A_{i_p})}. \quad (3)$$

Note that the column vector A_{i_p} is an implicit function of the shape parameter σ .

4 Numerical Example

We present some numerical results generated by applying a greedy algorithm based on incremental QR factorization to solve the 2D Poisson problem $\nabla^2 u = f$ in $\Omega \in [0, 1] \times [0, 1]$ subject to the Dirichlet boundary condition $u = g$ in $\partial\Omega$. The problem considered here has been taken from [10], where

$$\begin{aligned} f &= -\frac{751\pi^2}{144} \sin(\pi x/6) \sin(7\pi x/4) \sin(3\pi y/4) \sin(5\pi y/4) \\ &\quad + (7\pi^2/12) \cos(\pi x/6) \cos(7\pi x/4) \sin(3\pi y/4) \sin(5\pi y/4) \\ &\quad + (15\pi^2/8) \sin(\pi x/6) \sin(7\pi x/4) \cos(3\pi y/4) \cos(5\pi y/4), \\ g &= \sin(\pi x/6) \sin(7\pi x/4) \sin(3\pi y/4) \sin(5\pi y/4). \end{aligned} \quad (4)$$

For this problem, we used a dictionary of 441 Gaussian RBFs (i.e., $\phi = \exp(-\|x - c\|^2/\sigma^2)$) with centers corresponding to a uniform 21×21 grid. The solution error norm is computed on a set of 51×51 points. Benchmark results for this problem were first generated using the standard unsymmetric collocation scheme. The L_∞ error norm of the best solution (obtained after experimenting with various values of σ) is 1.03×10^{-6} . Note that this solution is obtained using all 441 basis functions in the dictionary.

The convergence trends of the greedy scheme incorporating automatic shape parameter tuning is shown in Figure 1. At each iteration of the greedy algorithm, the optimum value of the shape parameter of the chosen basis function is computed by minimizing the residual error as discussed in the previous section. The horizontal line in Figure 1 represents the L_∞ error norm of the solution obtained using the standard collocation scheme. It can be seen that the greedy approach allows for the possibility of achieving higher accuracy compared to the standard collocation scheme, particularly when σ is tuned for each basis function chosen from the dictionary. For example, the L_∞ norm of the solution error using 250 basis functions is 1.3×10^{-8} , which is around two orders of magnitude better than what is achievable using the standard collocation scheme. It was observed that the greedy algorithm has a tendency to choose RBF centers close to the boundary of the domain, since this is the region where the residual error tends to be highest; see Figure 2.

Our numerical studies suggest that the greedy approach is significantly more efficient than the standard collocation scheme, particularly when the termination criteria is met using a moderate number of basis functions and when a large number of collocation points is necessary for an accurate solution. For example, when a set of 51×51 collocation points is used to solve the Poisson problem discussed here, the standard collocation scheme requires roughly 35 seconds. In comparison, the greedy scheme gives a more accurate solution (using 250 basis functions) in around 7 seconds.

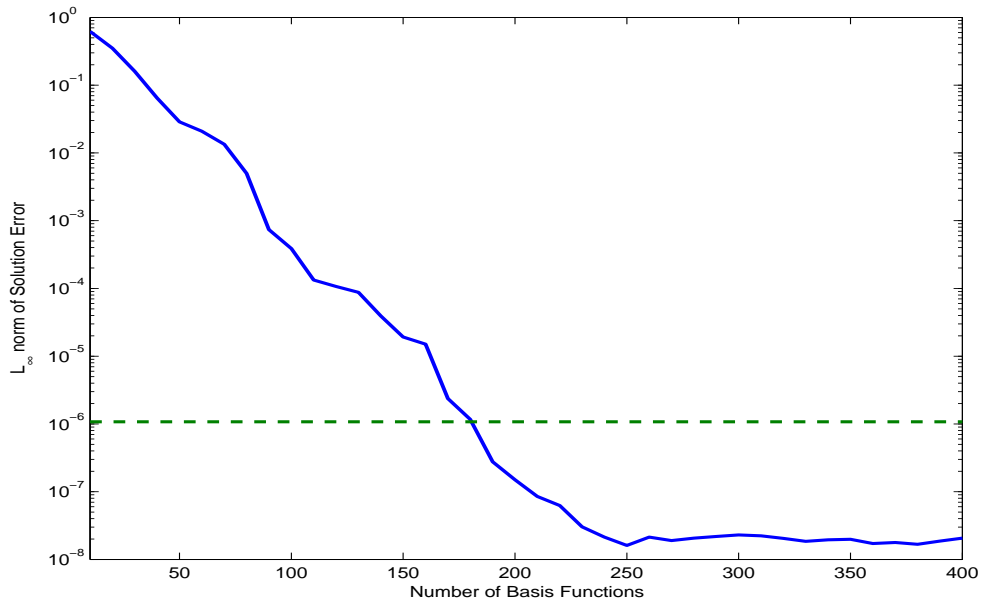


Figure 1: Convergence of L_∞ norm of solution error using the greedy algorithm. The dashed horizontal line represents the L_∞ error norm of the best solution obtained using the standard collocation scheme.

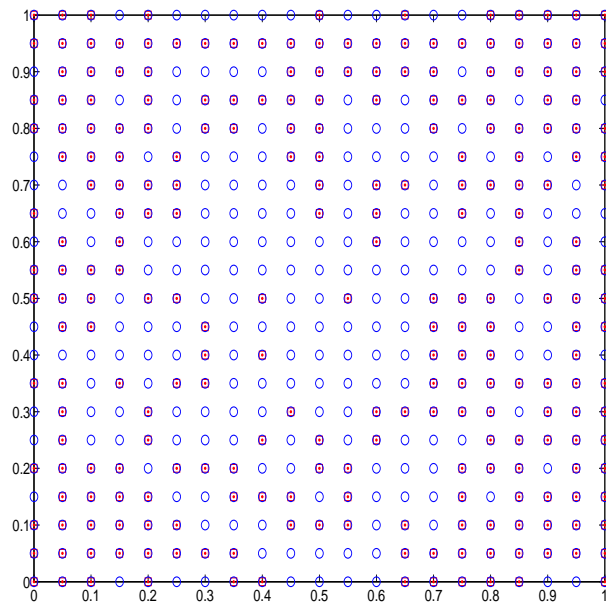


Figure 2: Spatial distribution of first 250 RBF centers chosen by the greedy algorithm.

5 Concluding Remarks

In this paper, we presented an outline of greedy algorithms for adaptive meshfree collocation of linear elliptic PDEs. The main idea was to adaptively select RBFs from a dictionary so as to minimize the residual error in the governing equations and the boundary conditions. The proposed approach based on incremental QR factorization allows stable

computation of the weights even when the collocation matrix is highly ill-conditioned. Further, only $O(np)$ memory and operations are incurred at iteration p .

As shown in this paper, it is possible to efficiently tune the shape parameter of each individual basis function by solving a one-dimensional minimization problem. In comparison, tuning the shape parameters within the framework of the standard unsymmetric collocation formulation is computationally infeasible, since that will involve the solution of a n -dimensional nonlinear programming problem. In this regard, the proposed greedy approach presents a significant enhancement to the standard unsymmetric collocation scheme, since it potentially allows the efficient and accurate solution of a wide class of operator problems without recourse to *expensive* numerical experiments to determine the optimum value of the shape parameter.

References

- [1] Kansa, E.J., "Multiquadrics - A Scattered Data Approximation Scheme with Applications to Computational Fluid Dynamics: I. Surface Approximations and Partial Derivative Estimates", *Comput. Math. Appl.*, 19(6-8), 1990, pp. 127–145.
- [2] Kansa, E.J., "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(6-8), 1990, pp. 147–161.
- [3] Fasshauer, G.E., "Solving Partial Differential Equations by Collocation with Radial Basis Functions," In: A. LeMhaut, C. Rabut and L. Schumaker, Editors, *Surface Fitting and Multiresolution Methods*, 1997.
- [4] Franke, C., and Schaback, R., "Solving Partial Differential Equations by Collocation Using Radial Basis Functions," *Appl. Math. Comp.*, 93, 1998, pp. 73–83.
- [5] Natarajan, B.K., "Sparse Approximate Solutions to Linear Systems," *SIAM J. of Computing*, 25(2), 1995, pp. 227–234.
- [6] Hon, Y.C., Schaback, R., and Zhou, X., "Adaptive Greedy Algorithm for Solving Large RBF Collocation Problems," *Numerical Algorithms*, 32, 2003, pp. 13–25.
- [7] Mallat, S., and Zhang, Z., "Matching Pursuit in a Time-Frequency Dictionary," *IEEE Trans. Signal Processing*, 41, 1993, pp. 3397–3415.
- [8] Nair, P.B., Choudhury, A., and Keane, A.J., "Some Greedy Learning Algorithms for Sparse Regression and Classification with Mercer Kernels," *J. Machine Learning Research*, 3, 2002, pp. 781–801.
- [9] Hon, Y.C., and Schaback, R., "On Unsymmetric Collocation by Radial Basis Functions," *Appl. Math. Comp.*, 119, 2001, pp. 177–186.
- [10] Cheng, A.H.-D., Golberg, M.A., Kansa, E.J. and Zammito, G., "Exponential convergence and h-c multiquadric collocation method for partial differential equations," *Num. Meth. Partial Differential Equations*, 19(5), 2003, pp. 571–594.