Journal of Computational Physics 331 (2017) 137–156

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Stable computations with flat radial basis functions using vector-valued rational approximations

Grady B. Wright [a,*], Bengt Fornberg [b]

[a] Department of Mathematics, Boise State University, Boise, ID 83725-1555, USA
[b] Department of Applied Mathematics, University of Colorado, 526 UCB, Boulder, CO 80309, USA

ARTICLE INFO

Article history:
Received 15 October 2016
Accepted 21 November 2016
Available online 25 November 2016

Keywords:
RBF
Shape parameter
Ill-conditioning
Contour-Padé
RBF-QR
RBF-GA
Rational approximation
Common denominator
RBF-FD
RBF-HFD

ABSTRACT

One commonly finds in applications of smooth radial basis functions (RBFs) that scaling the kernels so they are 'flat' leads to smaller discretization errors. However, the direct numerical approach for computing with flat RBFs (RBF-Direct) is severely ill-conditioned. We present an algorithm for bypassing this ill-conditioning that is based on a new method for rational approximation (RA) of vector-valued analytic functions with the property that all components of the vector share the same singularities. This new algorithm (RBF-RA) is more accurate, robust, and easier to implement than the Contour-Padé method, which is similarly based on vector-valued rational approximation. In contrast to the stable RBF-QR and RBF-GA algorithms, which are based on finding a better conditioned base in the same RBF-space, the new algorithm can be used with any type of smooth radial kernel, and it is also applicable to a wider range of tasks (including calculating Hermite type implicit RBF-FD stencils). We present a series of numerical experiments demonstrating the effectiveness of this new method for computing RBF interpolants in the flat regime. We also demonstrate the flexibility of the method by using it to compute implicit RBF-FD formulas in the flat regime and then using these for solving Poisson's equation in a 3-D spherical shell.

1. Introduction

Meshfree methods based on smooth radial basis functions (RBFs) are finding increasing use in scientific computing as they combine high order accuracy with enormous geometric flexibility in applications such as interpolation and for numerically solving PDEs. In these applications, one finds that the best accuracy is often achieved when their shape parameter ε is small, meaning that they are relatively flat [1,2].

The so called Uncertainty Principle, formulated in 1995 [3], has contributed to a widespread misconception that flat radial kernels unavoidably lead to numerical ill-conditioning. This 'principle' mistakenly assumes that RBF interpolants need to be computed by solving the standard RBF linear system (often denoted RBF-Direct). However, it has now been known for over a decade [4–7] that the ill-conditioning issue is specific to this RBF-Direct approach, and that it can be avoided using alternative methods. Three distinctly different numerical algorithms have been presented thus far in the literature for avoiding this ill-conditioning and thus open up the complete range of ε that can be considered. These are the Contour-Padé (RBF-CP) method [8], the RBF-QR method [9–11], and the RBF-GA method [12]. The present paper develops a new stable algorithm that is in the same category as RBF-CP.

* Corresponding author.
E-mail addresses: gradywright@boisestate.edu (G.B. Wright), fornberg@colorado.edu (B. Fornberg).

http://dx.doi.org/10.1016/j.jcp.2016.11.030

Table 1
Examples of analytic radial kernels featuring a shape-parameter ε that the RBF-RA procedure is immediately applicable to. The first three kernels are positive-definite and the last is conditionally negative definite.

Name	Abbreviation	Definition
Gaussian	GA	$\phi_\varepsilon(r) = e^{-(\varepsilon r)^2}$
Inverse quadratic	IQ	$\phi_\varepsilon(r) = 1/(1+(\varepsilon r)^2)$
Inverse multiquadric	IMQ	$\phi_\varepsilon(r) = 1/\sqrt{1+(\varepsilon r)^2}$
Multiquadric	MQ	$\phi_\varepsilon(r) = \sqrt{1+(\varepsilon r)^2}$

For fixed numbers of interpolation nodes and evaluations points, an RBF interpolant can be viewed as a vector-valued function of ε [8]. The RBF-CP method exploits the analytic nature of this vector-valued function in the complex ε-plane to obtain a vector-valued rational approximation that can be used as a proxy for computing stably in the $\varepsilon \to 0$ limit. One key property that is utilized in this method is that all the components of the vector-valued function share the same singularities (which are limited to poles). The RBF-CP method obtains a vector-valued rational approximation with this property from contour integration in the complex ε-plane and Padé approximation. However, this method is somewhat computationally costly and can be numerically sensitive to the determination of the poles in the rational approximations. In this paper, we follow a similar approach of generating vector-valued rational approximants, but use a newly developed method for computing these. The advantages of this new method, which we refer to as RBF-RA, over RBF-CP include:

- Significantly higher accuracy for the same computational cost.
- Shorter, simpler code involving fewer parameters, and less use of complex floating point arithmetic.
- More robust algorithm for computing the poles of the rational approximation.

As with the RBF-CP method, the new RBF-RA method is limited to a relatively low number of interpolation nodes (just under a hundred in 2-D, a few hundred in 3-D), but is otherwise more flexible than RBF-QR and RBF-GA in that it immediately applies to any type of smooth RBFs (see Table 1 for examples), to any dimension, and to more generalized interpolation techniques, such as appending polynomials to the basis, Hermite interpolation, and customized matrix-valued kernel interpolation. Additionally, it can be immediately applied to computing RBF generated finite difference formulas (RBF-FD) and Hermite (or compact or implicit) RBF-FD formulas (termed RBF-HFD), which are based on standard and Hermite RBF interpolants, respectively [13]. RBF-FD formulas have seen tremendous applications to solving various PDEs [13–22] since being introduced around 2002 [23,24]. It is for computing RBF-FD and RBF-HFD formulas that we see the main benefits of the RBF-RA method, as these formulas are typically based on node sizes well within its limitations. Additionally, in the case of RBF-HFD formulas, the RBF-QR and RBF-GA methods cannot be readily used.

Another two areas where RBF-RA is applicable is in the RBF partition of unity (RBF-PU) method [25–28] and domain decomposition [29,30], as these also involve relatively small node sets. While the RBF-QR and RBF-GA methods are also suitable for these problems, they are limited to the Gaussian (GA) kernel, whereas the RBF-RA method is not. In the flat limit, different kernels sometimes give results of different accuracies. It is therefore beneficial to have stable algorithms that work for all analytic RBFs. Figure 8 in [8] shows an example where the flat limits of multiquadric (MQ) and inverse quadratic (IQ) interpolants are about two orders of magnitude more accurate than for GA interpolants.

The remainder of the paper is organized as follows. We review the issues with RBF interpolation using flat kernels in Section 2. We then discuss the new vector-valued rational approximation method that forms the foundation for the RBF-RA method for stable computations in Section 3. Section 4 describes the analytic properties of RBF interpolants in the complex ε-plane and how the new rational approximation method is applicable to computing these interpolants and also to computing RBF-HFD weights. We present several numerical studies in Section 5. The first of these focuses on interpolation and illustrates the accuracy and robustness of the RBF-RA method over the RBF-CP method and also compares these methods to results using multiprecision arithmetic. The latter part of Section 5 focuses on the application of the RBF-RA method to generating RBF-HFD formulas for the Laplacian and contains results from applying these formulas to solving Poisson's equation in a 3-D spherical shell. We make some concluding remarks about the method in Section 6. Finally, a brief MATLAB code is given in Appendix A and some suggestions on one main free parameters of the algorithms are given in Appendix B.

2. The nature of RBF ill-conditioning in the flat regime

For notational simplicity, we will first focus on RBF interpolants of the form

$$s(\mathbf{x},\varepsilon) = \sum_{i=1}^{N} \lambda_i \phi_\varepsilon(\|\mathbf{x} - \hat{\mathbf{x}}_i\|), \tag{1}$$

where $\{\hat{\mathbf{x}}_i\}_{i=1}^N \subset \mathbb{R}^d$ are the interpolation nodes (or centers), $\mathbf{x} \in \mathbb{R}^d$ is some evaluation point, $\|\cdot\|$ denotes the two-norm, and ϕ_ε is an analytic radial kernel that is positive (negative) definite or conditionally positive (negative) definite (see Table 1