Approximate Moving Least-Squares Approximation: A Fast and Accurate Multivariate Approximation Method

Gregory E. Fasshauer

Abstract. We propose a fast and accurate approximation method for large sets of multivariate data using radial functions. In the traditional radial basis function approach this task is usually accomplished by solving a large system of linear equations stemming from an interpolation formulation. In the traditional moving least-squares method one needs to solve a small linear system for each evaluation of the approximant. We present an approximation scheme – based on the work on approximate approximation by Maz'ya and Schmidt – that has approximation properties similar to the moving least-squares method, but completely avoids the solution of linear systems. Moreover, the sums required for the evaluation of the approximant can be processed quickly. We establish a connection to traditional radial basis function approximation by using appropriate radial generating functions. Examples of locally supported as well as globally supported functions with arbitrary approximation orders are given.

§1. Introduction

In this paper we propose a fast and accurate approximation method for large sets of multivariate data of the form $\{(x_i, f(x_i)) \ : \ i = 1, \ldots, N\} \subset \mathbb{R}^s \times \mathbb{R}$ by radial functions. In the traditional radial basis function (RBF) approach (see, e.g., [12,13]) this task is usually accomplished by *interpolation* of the data which in turn leads to a large system of linear equations. The zero-structure (sparse or dense) of the system matrix will depend on the choice of radial function. Moreover, once the expansion coefficients have been determined, subsequent evaluation of the RBF interpolant will require evaluation of a sum of the form

$$\mathcal{I}f(x) = \sum_{j=1}^{N} c_j \phi(\|x - x_j\|), \qquad x \in \mathbb{R}^s .$$

Curve and Surface Fitting: Saint-Malo 2002
XXX (eds.), pp. 1–10.

ISBN XXX.

In the traditional moving least-squares (MLS) method [9] the amount of work is shifted. There is no large system to solve. Instead, for every evaluation one needs to solve a small linear system (normal equations) to find the coefficients of the moving local (polynomial) approximant, and then evaluate a summation (more details are given in Sect. 2). It is our goal (started in [4]) to devise an approximation scheme that has approximation properties similar to the MLS method, but completely avoids the solution of linear systems. Moreover, we want to be able to process the sums required for the evaluation of the approximant quickly.

In this paper we present such a method – based on the work on *approximate approximation* by Maz'ya and Schmidt [11]. We establish a connection to traditional RBF approximation by using appropriate radial generating functions. Examples of generating functions with arbitrary approximation orders are given. The final section outlines a fast summation approach applicable to general global methods with rapidly decaying generating functions.

Throughout the paper we will be using multi-index notation. We call $\alpha = (\alpha_1, \ldots, \alpha_s) \in \mathbb{N}^s$ a multi-index with length $|\alpha| = \sum_{i=1}^{s} \alpha_i$. The multivariate factorial is defined by $\alpha! = \alpha_1! \cdots \alpha_s!$. If $x \in \mathbb{R}^s$, then the monomials are $x^\alpha = x_1^{\alpha_1} \cdots x_s^{\alpha_s}$. Multivariate differential operators are denoted by $D^\alpha = \partial_1^{\alpha_1} \partial_2^{\alpha_2} \cdots \partial_s^{\alpha_s}$, where ∂_i denotes differentiation with respect to the i-th coordinate direction in $\mathbb{R}^s$. Finally, $\alpha \leq p$, for some integer p if $\alpha_i \leq p$ for all $i = 1, \ldots, s$.

§2. Moving Least-Squares via Constrained Optimization

We begin the motivation of our method with a brief review of the so-called Backus-Gilbert approach to MLS approximation (a more detailed discussion covering also the more familiar discrete least-squares formulation can be found in [4]). The basic idea is to approximate the given data with a quasi-interpolant of the form

$$\mathcal{Q}f(x) = \sum_{j=1}^{N} f(x_j)\Psi_j(x), \qquad x \in \mathbb{R}^s\,. \tag{1}$$

The generating functions Ψ_j are determined such that

$$\sum_{j=1}^{N} p(x_j)\Psi_j(x) = p(x), \qquad \forall p \in \Pi_d^s,$$

and

$$\frac{1}{2}\sum_{j=1}^{N} \Psi_j^2(x) w_j(x) \to \min.$$

Here the w_j are considered to be positive weight functions. The first of these constraints guarantees reproduction of polynomials up to degree d, and thus approximation order $d+1$. Such a theorem was proven by both Levin [10] and Wendland [16].

From the proofs of the approximation order result in [10] and [16], a solution procedure is suggested as follows. We use Lagrange multipliers and identify $\Phi_j(x) = \dfrac{1}{w_j(x)}$. This leads to the following formula for the generating functions:

$$\Psi_j(x) = \left(\sum_{k=1}^{m} \lambda_k(x) p_k(x_j) \right) \Phi_j(x), \qquad j = 1, \ldots, N, \tag{2}$$

where the Lagrange multipliers $\lambda_k(x)$ are the solution of

$$G\lambda = q, \qquad \text{with } q = [p_1(x), \ldots, p_m(x)]^T. \tag{3}$$

The matrix G is a Gram matrix with

$$G_{k,\ell}(x) = \sum_{j=1}^{N} p_k(x_j) p_\ell(x_j) \Phi_j(x).$$

The best known (and simplest) example of an MLS method is Shepard's method [14] which arises in the case $d = 0$. In this case

$$\mathcal{Q}f(x) = \sum_{j=1}^{N} f(x_j) \frac{\Phi_j(x)}{\sum_{k=1}^{N} \Phi_k(x)}. \tag{4}$$

The approximation order result quoted above guarantees that this method has approximation order $\mathcal{O}(h)$ provided Φ_j has support size $\rho_j = ch$.

The following is an implication of the approximation order result by Levin and Wendland.

Corollary 1. *The choice of weight functions in MLS approximation does not affect the approximation order.*

This was illustrated numerically in [4] for Shepard's method based on three different locally supported weights. The convergence rate was approximately the same for all three methods – but significantly better than linear (approximately $\mathcal{O}(h^{1.3})$). We will comment on this fact later.

§3. Matrix-free Formulations

As indicated in [4], there seem to be at least three approaches to obtaining a matrix-free formulation for radial basis function approximation. Of those three approaches, the most promising line of attack seems to be the direct construction of appropriate generating functions. That is what we want to pursue here. This approach was also suggested in [5]. In order to be able to motivate our construction, we recall some formulas for the case $s = 2$ and $d = 1$ from [4].

Example 1. According to (2) and (3), the generating functions are given by

$$\Psi_j(x,y) = [\lambda_1(x,y) + \lambda_2(x,y)(x_j - x) + \lambda_3(x,y)(y_j - y)]\,\Phi_j(x,y), \quad (5)$$

where x and y denote the components of a point in $\mathbb{R}^2$ and

$$\begin{aligned}\lambda_1(x,y) =& \frac{1}{D}\left[\mu_{11}^2 - \mu_{20}\mu_{02}\right],\\ \lambda_2(x,y) =& \frac{1}{D}\left[\mu_{10}\mu_{02} - \mu_{01}\mu_{11}\right], \quad (6)\\ \lambda_3(x,y) =& \frac{1}{D}\left[\mu_{20}\mu_{01} - \mu_{10}\mu_{11}\right],\end{aligned}$$

with

$$D = \mu_{10}^2\mu_{02} + \mu_{20}\mu_{01}^2 - \mu_{00}\mu_{20}\mu_{02} - 2\mu_{10}\mu_{01}\mu_{11} + \mu_{00}\mu_{11}^2$$

and discrete moments

$$\mu_\alpha = \sum_{j=1}^{N}(x_j - x)^\alpha \Phi_j(x), \qquad x \in \mathbb{R}^s, \qquad |\alpha| \le 2d.$$

This guarantees that we can approximate the given data by an expansion of the form (1) with approximation order $\mathcal{O}(h^{d+1}) = \mathcal{O}(h^2)$.

§4. Moment Conditions

Inspecting (6), we see that formula (5) for Ψ_j can be turned into the trivial identity $\Psi_j = \Phi_j$ by enforcing a certain set of conditions on the moments μ_α. For the example given above, these are

$$\begin{aligned}\mu_{10} =& \sum_{j=1}^{N}(x_j - x)\Phi_j(x,y) = 0,\\ \mu_{01} =& \sum_{j=1}^{N}(y_j - y)\Phi_j(x,y) = 0.\end{aligned}$$

Then it is easy to see that

$$\Psi_j(x,y) = \frac{\Phi_j(x,y)}{\sum_{j=1}^{N}\Phi_j(x,y)},$$

which now makes (1) look just like the basic Shepard method (4). However, contrary to the approximation order predicted by the traditional MLS theory, we have achieved an increase in approximation order by imposing additional constraints on the weight functions. Here lies the explanation why all three Shepard methods in the example from [4] mentioned at the end of Sect. 2 performed better than linear. The preceding arguments can be generalized:

Proposition 2. *Let $\Omega \subset \mathbb{R}^s$. If $f \in C^{d+1}(\Omega)$, $\{x_i : i = 1, \ldots, N\} \subset \Omega$ are quasi-uniformly distributed with "mesh size" h, the generating functions $\Psi_j(x) = \Psi(x - x_j)$ are compactly supported with support size $\rho_j = ch$ ($c = \text{const.}$), and the* discrete moment conditions

$$\mu_\alpha = \sum_{j=1}^{N} (x_j - x)\Psi_j(x) = \delta_{\alpha 0}, \qquad 0 \le |\alpha| \le d,$$

are satisfied, then

$$\mathcal{Q}_h f(x) = \sum_{j=1}^{N} f(x_j)\Psi\left(\frac{x - x_j}{\rho_j}\right)$$

has approximation order $\mathcal{O}(h^{d+1})$.

We can interpret the effect of the moment conditions in Proposition 2 as follows. The generating function Ψ can be viewed as

$$\Psi(x) = \Phi_0(x)q(x), \qquad q \in \Pi^s_{d+1},$$

with Φ_0 a new (arbitrary) weight function, and q an s-variate polynomial orthogonal with respect to Φ_0. Unfortunately, constructing such a function Ψ is a very difficult problem. It would require discrete multivariate orthogonal polynomials at scattered centers.

§5. Approximate Approximation

The key to a numerically feasible approximation scheme is given by a series of papers by Maz'ya and Schmidt in which they establish what they coined approximate approximation. Their most relevant result (see [11]) is summarized as

Theorem 3. (Maz'ya & Schmidt) *Let $f \in C^{d+1}(\mathbb{R}^s)$, $\{x_\nu : \nu \in \mathbb{Z}^s\} \subset \mathbb{R}^s$, and Ψ a continuous generating function which satisfies the* continuous moment conditions

$$\int_{\mathbb{R}^s} x^\alpha \Psi(x)dx = \delta_{\alpha 0}, \qquad 0 \le |\alpha| \le d, \tag{7}$$

along with a mild decay requirement. Then

$$\mathcal{M}_h f(x) = \mathcal{D}^{-s/2} \sum_{\nu \in \mathbb{Z}^s} f(x_\nu)\Psi\left(\frac{x - x_\nu}{\sqrt{\mathcal{D}}h}\right)$$

leads to

$$\|\mathcal{M}_h f - f\|_\infty = \mathcal{O}(h^{d+1} + \varepsilon_0(\Psi, \mathcal{D})).$$

The statement of Theorem 3 tells us that we can approximate with *numerical approximation order* $\mathcal{O}(h^{d+1})$ until a certain saturation error ε_0 takes over. The role of the parameter $\mathcal{D}$ is to regulate when this is to occur. In other words, by scaling the generating functions appropriately, we can assure that no saturation occurs within the range of machine accuracy on a given computer.

We point out that Theorem 3 is formulated for gridded data on an unbounded domain. However, similar results for scattered data on bounded domains can be found in the work of Maz'ya and Schmidt. The numerical examples referred to later on are so far restricted to the gridded setting. We have not yet implemented a practical method for scattered data.

§6. Construction of Generating Functions

To make the connection to RBF approximation, we assume the generating function is radial, i.e., $\Psi(x) = q(\|x\|^2)\phi_0(\|x\|^2)$. Therefore, we want (see (7))

$$\int_{\mathbb{R}^s} \|x\|^{2k} q(\|x\|^2)\phi_0(\|x\|^2)dx = \delta_{k0}, \qquad 0 \le k \le d, \tag{8}$$

to ensure approximation order $\mathcal{O}(h^{2d+2})$.

By using s-dimensional spherical coordinates and the subsequent change of variables $y = r^2$, we see that (8) is equivalent to

$$\frac{\pi^{s/2}}{\Gamma(s/2)} \int_0^\infty y^{k-1} q(y)\phi_0(y) y^{s/2} dy = \delta_{k0}, \qquad 0 \le k \le d. \tag{9}$$

This tells us that we need to find a *univariate* polynomial q which is orthogonal with respect to the weight $y^{s/2}\phi_0(y)$. Thus, the guiding principle in constructing a generating function with the desired approximation order is to pick an (arbitrary) *univariate weight function* ϕ_0, and then compute the *multivariate generating function* via the 1D moment conditions (9).

We have done this for compactly supported as well as globally supported weight functions ϕ_0.

If $\phi_0(r) = (1 - \sqrt{r})_+^4(4\sqrt{r} + 1)$, then we are able to compare our approximate approximation results to interpolation with the compactly supported RBF $\phi(r) = (1 - r)_+^4(4r + 1)$ introduced by Wendland [15]. Corresponding generating functions for $s = 1, 2, 3$ and approximation orders 2, 4, and 6 are listed in [5], and numerical comparisons between RBF interpolation, MLS approximation and approximate MLS approximation were also reported there.

Similarly, we can compute globally supported generating functions for $s = 1, 2, 3$ with approximation orders 2, 4, or 6 by starting, e.g., with $\phi_0(r) = e^{-r}$. This leads to the functions listed in Table 1.

s	$\mathcal{O}(h^2)$	$\mathcal{O}(h^4)$	$\mathcal{O}(h^6)$
1	$\frac{1}{\sqrt{\pi}} e^{-\|x\|^2}$	$\frac{1}{\sqrt{\pi}} \left(\frac{3}{2} - \|x\|^2\right) e^{-\|x\|^2}$	$\frac{1}{\sqrt{\pi}} \left(\frac{15}{8} - \frac{5}{2}\|x\|^2 + \frac{1}{2}\|x\|^4\right) e^{-\|x\|^2}$
2	$\frac{1}{\pi} e^{-\|x\|^2}$	$\frac{1}{\pi} \left(2 - \|x\|^2\right) e^{-\|x\|^2}$	$\frac{1}{\pi} \left(3 - 3\|x\|^2 + \frac{1}{2}\|x\|^4\right) e^{-\|x\|^2}$
3	$\frac{1}{\pi^{3/2}} e^{-\|x\|^2}$	$\frac{1}{\pi^{3/2}} \left(\frac{5}{2} - \|x\|^2\right) e^{-\|x\|^2}$	$\frac{1}{\pi^{3/2}} \left(\frac{35}{8} - \frac{7}{2}\|x\|^2 + \frac{1}{2}\|x\|^4\right) e^{-\|x\|^2}$

Tab. 1. Globally supported generating functions.

More generally, if we let $L_d^{s/2}$ be the generalized Laguerre polynomial of degree d, then

$$\Psi_j(x) = \frac{1}{\pi^{s/2}} L_d^{s/2} \left(\frac{\|x - x_j\|^2}{\mathcal{D}h^2}\right) \exp\left(-\frac{\|x - x_j\|^2}{\mathcal{D}h^2}\right)$$

has approximation order $\mathcal{O}(h^{2d+2})$ in $\mathbb{R}^s$.

§7. Fast Summation via Taylor Expansions

Once we have determined suitable generating functions for the quasi-interpolant (1), the remaining task is to evaluate the corresponding sum efficiently. Based on our experience with the generating functions mentioned in Sect. 6, better results are achieved with the globally supported functions of Table 1. In the traditional RBF setting, Beatson and coworkers (see, e.g., [1,2]) have developed fast evaluation methods especially for polyharmonic splines and multiquadrics. Our work on this subject is motivated by the general fast tree code algorithms of Krasny and coworkers (see, e.g., [3]), and by the fast Gauss transform of Greengard and Strain [8].

The main idea is to divide the computational mesh into clusters C (containing the source points or centers). One then computes Taylor series expansions about the centers of the source clusters. In order to devise a fast summation algorithm, one needs fast decay of the generating functions and a recurrence relation for the Taylor coefficients. The following theorem can be applied for any quasi-interpolant.

Theorem 4. *Let I_c be the index set denoting the sources x_j which lie in a cluster C with center x_c. Then the value of the quasi-interpolant at x is given by*

$$\mathcal{Q}f(x) = \sum_c \sum_{j \in I_c} f(x_j) \Psi(x - x_j).$$

Moreover

$$\mathcal{Q}f(x) = \sum_c \sum_{\alpha \geq 0} T_\alpha(x, x_c) m_\alpha(c),$$

with the multivariate moments $m_\alpha(c)$ *given by*

$$m_\alpha(c) = \sum_{j \in I_c} f(x_j)(x_j - x_c)^\alpha,$$

and $T_\alpha(x, x_c) = \dfrac{1}{\alpha!} D^\alpha \Psi(z)|_{z=x-x_c}$.

In order to devise a fast tree code one uses either the truncated Taylor expansion

$$\mathcal{Q}f(x) = \sum_c \sum_{|\alpha|<p} T_\alpha(x, x_c) m_\alpha(c),$$

or direct summation

$$\mathcal{Q}f(x) = \sum_c \sum_{j \in I_c} f(x_j) \Psi(x - x_j),$$

depending on which is more efficient for the given cluster size and desired accuracy, i.e., p is the minimum order satisfying our accuracy criterion.

If we take $\Psi(x) = e^{-\|x\|^2}$, then

$$T_\alpha(x, y) = \frac{1}{\alpha!} D_y^\alpha e^{-\|x-y\|^2} = \frac{(-1)^{|\alpha|}}{\alpha!} h_\alpha(x - y),$$

and we obtain the expansions underlying the fast Gauss transform of Greengard and Strain [8]. Here h_α are multivariate Hermite functions. As mentioned earlier, the well-known recurrence relation for Hermite functions adds to the efficiency of the fast Gauss transform.

Initial experiments comparing an implementation of the fast Gauss transform taken from [7] to direct summation in the case $s = 1$ have shown an approximate 300-fold speedup. Thus, we are able to approximate data given at 250,000 points in roughly the same time it takes us for direct summation at 500 points. In other words, investing the same amount of time, the accuracy can be improved by roughly five orders of magnitude.

In order to obtain a similar fast summation algorithm for our generating functions of Table 1, we need to derive recurrence relations for the higher-order Gauss-Laguerre functions. We close with two such examples.

Example 2. We take $\Psi(x) = e^{-\|x\|^2}$. Then the Taylor coefficients are

$$b_\alpha = \frac{1}{\alpha!} D^\alpha \Psi(x).$$

Now, using the multivariate Leibniz rule, we get the recurrence relation

$$b_\alpha + \frac{2}{\alpha_\ell} x_\ell b_{\alpha - e_\ell} + \frac{2}{\alpha_\ell} b_{\alpha - 2e_\ell} = 0, \qquad \ell = 1, \ldots, s.$$

Example 3. If $\Psi(x) = \left(\frac{s+2}{2} - \|x\|^2\right) e^{-\|x\|^2}$, then the Taylor coefficients are

$$a_\alpha = \frac{1}{\alpha!} D^\alpha \Psi(x),$$

and (again using the multivariate Leibniz rule) we get the nested recursion

$$a_\alpha + \frac{2}{\alpha_\ell} x_\ell a_{\alpha - e_\ell} + \frac{2}{\alpha_\ell} a_{\alpha - 2e_\ell} = b_\alpha, \qquad \ell = 1, \ldots, s.$$

with b_α as in the previous example.

§8. Concluding Remarks

We have proposed a computationally efficient (and accurate) meshfree method. We have also seen that the choice of MLS weight function Φ (with vanishing moments) has considerable effect on the rate of convergence. Our numerical experiments (not reported here) have shown that local approximate MLS approximation seems less stable and more sensitive to the choice of the scale parameter $\mathcal{D}$ of Theorem 3. In order to get good results with local functions the support has to be made essentially global. A multilevel technique such as frequently used for interpolation with compactly supported RBFs does not help (see [5]). The fast summation technique based on Taylor expansions suggested here applies to arbitrary radial functions. The fast tree code algorithm can be made even more efficient by using a cluster-cluster algorithm such as in the fast multipole method. A paper containing numerical experiments as well as more details on the fast summation method is in preparation [6].

Acknowledgments. Supported by the National Science Foundation under grant DMS-0073636.

References

1. Beatson, R. K., J. B. Cherrie, and C. T. Mouat, Fast fitting of radial basis functions: Methods based on preconditioned GMRES iteration, Advances in Comp. Math. **11** (1999), 253–270.
2. Beatson, R. K., W. A. Light, and S. Billings, Fast solution of the radial basis function interpolation methods: Domain decomposition methods, SIAM J. Scient. Computing **22** (2000), 1717–1740.
3. Duan, Z.-H. and R. Krasny, An Ewald summation based multipole method, J. Chemical Phys. **113** (2000), 3492–3495.
4. Fasshauer, G. E., Matrix-free multilevel moving least-squares methods, in *Approximation Theory X: Splines, Wavelets, and Applications,*

C. K. Chui, L. L. Schumaker, and J. Stöckler (eds), Vanderbilt University Press, Nashville, 2002, 271–281.

5. Fasshauer, G. E., Approximate moving least-squares approximation with compactly supported weights, in *Lecture Notes in Computer Science and Engineering Vol.26: Meshfree Methods for Partial Differential Equations*, M. Griebel and M. A. Schweitzer (eds), Springer, Berlin, 2002, 105–116.
6. Fasshauer, G. E., High-order moving least-squares approximation via fast radial Laguerre transforms, in preparation, 2002.
7. Florence, A. G. and C. F. van Loan, A Kronecker product formulation of the fast Gauss transform, preprint, 2000.
8. Greengard, L. and J. Strain, The fast Gauss transform, SIAM J. Sci. Statist. Comput. **12** (1991), 79–94.
9. Lancaster, P. and K. Šalkauskas, Surfaces generated by moving least squares methods, Math. Comp. **37** (1981), 141–158.
10. Levin, D., The approximation power of moving least-squares, Math. Comp. **67** (1998), 1517–1531.
11. Maz'ya, V. and G. Schmidt, On quasi-interpolation with non-uniformly distributed centers on domains and manifolds, J. Approx. Theory **110** (2001), 125–145.
12. Powell, M. J. D., The theory of radial basis functions in 1990, in *(Advances in Numerical Analysis II: Wavelets, Subdivision, and Radial Basis Functions)*, W. Light (ed.), Oxford University Press, Oxford, 1992, 105–210.
13. Schaback, R., On the efficiency of interpolation by radial basis functions, in *Surface Fitting and Multiresolution Methods,* A. LeMéhauté, C. Rabut, and L. L. Schumaker (eds), Vanderbilt University Press, Nashville TN, 1997, 309–318.
14. Shepard, D., A two dimensional interpolation function for irregularly spaced data, Proc. 23rd Nat. Conf. ACM, 1968, 517–523.
15. Wendland, H., Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree, Advances in Comp. Math. **4** (1995), 389–396.
16. Wendland, H., Local polynomial reproduction and moving least squares approximation, IMA J. Numer. Anal. **21** (2001), 285–300.

Greg Fasshauer
Department of Applied Mathematics
Illinois Institute of Technology
Chicago, IL 60616
fass@amadeus.math.iit.edu
http://amadeus.math.iit.edu/∼fass