

Toward Approximate Moving Least Squares Approximation with Irregularly Spaced Centers

Gregory E. Fasshauer¹

*Department of Applied Mathematics,
Illinois Institute of Technology, Chicago, IL 60616, U.S.A.*

Abstract

By combining the well known moving least squares approximation method and the theory of approximate approximations due to Maz'ya and Schmidt we are able to present an approximate moving least squares method which inherits the simplicity of Shepard's method along with the accuracy of higher-order moving least squares approximations. In this paper we focus our interest on practical implementations for irregularly spaced data sites. The two schemes described here along with some first numerical experiments are to be viewed as exploratory work only. These schemes apply to centers that are obtained from gridded centers via a smooth parametrization. Further work to find a robust numerical scheme applicable to arbitrary scattered data is needed.

Key words: moving least squares, approximate approximation, irregularly spaced centers

PACS: 02.30.Mv, 02.60.-x

Email address: fasshauer@iit.edu (Gregory E. Fasshauer).

URL: <http://amadeus.math.iit.edu/~fass> (Gregory E. Fasshauer).

¹ Research supported by the National Science Foundation under grant DMS-0073636.

1 Introduction

In a series of recent papers [1]–[4] we have investigated the combination of moving least squares approximation (MLS) (see, e.g., [5]) and approximate approximation (see, e.g., [8]). This combination (which we call approximate MLS approximation) yields an efficient and accurate approximation method for multivariate data that gets by without the solution of linear systems. So far – even though the fundamental theoretical results by Maz’ya and Schmidt on approximate approximation [8] include the case of irregularly spaced data sites – we have focussed our attention and computational experiments on the easier case of regularly spaced centers. In this paper we present two different approaches for certain kinds of irregularly spaced data that are designed to maintain the approximation orders achieved in the regular case.

Specifically, we are interested in approximating large sets of multivariate data of the form $\{(\mathbf{x}_i, f(\mathbf{x}_i)) : i = 1, \dots, N\} \subset \mathbb{R}^s \times \mathbb{R}$ by approximate moving least squares approximants with radial weights $\frac{1}{\Phi_j} = \frac{1}{\phi(\|\cdot - \mathbf{x}_j\|)}$. As in our earlier papers we are interested in methods that are

- matrix-free
- and have high approximation order.

It is well known that Shepard’s method

$$\mathcal{S}f(\mathbf{x}) = \sum_{j=1}^N f(\mathbf{x}_j) \frac{\Phi_j(\mathbf{x})}{\sum_{k=1}^N \Phi_k(\mathbf{x})}, \quad \mathbf{x} \in \mathbb{R}^s, \quad (1)$$

satisfies the first of our criteria, but has approximation order $\mathcal{O}(h)$ only. Below we will summarize our earlier results that established “Shepard-like” methods with higher approximation order on regularly spaced data. This will lead to approximants of the general form

$$\mathcal{Q}f(\mathbf{x}) = \sum_{j=1}^N f(\mathbf{x}_j) \Psi_j(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^s. \quad (2)$$

This kind of expansion – in the general case when the generating functions Ψ_j do not satisfy the cardinality (or delta-function) property $\Psi_j(\mathbf{x}_i) = \delta_{ij}$ – is usually referred to as a *quasi-interpolant* (otherwise as a *cardinal interpolant*). By imposing certain restrictions (moment conditions) on the generating functions, the approximation order of these methods can be higher than that of Shepard’s method (where the only requirement on the Ψ_j is that they form a partition of unity). Clearly, quasi-interpolants are very efficient approximation schemes, as the expansion coefficients are directly given by the data, and the only computational task is evaluation of the sum (2). Efficiency in accomplish-

ing this task is a difficult and important problem in its own right. We will not focus on that problem here. Some basic ideas, however, have been reported in [4]. Instead we focus on the modification of our earlier methods for the case of irregularly spaced data.

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

2 Approximate Moving Least Squares Approximation

2.1 Moving Least Squares Approximation

Moving least-squares methods are at the basis of numerous meshfree approximation methods for the solution of partial differential equations that have recently been suggested by practitioners as an alternative to the traditional finite element method (see, e.g., [7] and references therein).

As mentioned in the introduction, we assume to have data $\{(\boldsymbol{x}_i, f(\boldsymbol{x}_i))\}_{i=1}^N \subset \mathbb{R}^s \times \mathbb{R}$ with distinct data sites \boldsymbol{x}_i and f some (smooth) function, and we want to approximate them with an expansion of the form (2). Our goal is to determine the functions $\Psi_j = \Psi(\cdot - \boldsymbol{x}_j)$ in such a way that (2) can be computed efficiently and has high approximation order. In the MLS literature the latter is ensured by enforcing polynomial reproduction, i.e.,

$$\sum_{j=1}^N p(\boldsymbol{x}_j) \Psi_j(\boldsymbol{x}) = p(\boldsymbol{x}), \quad \text{for all } p \in \Pi_d^s. \quad (3)$$

Here Π_d^s is the space of s -variate polynomials of total degree at most d with $\dim \Pi_d^s = \binom{s+d}{d} =: m$. The values of the generating functions $\Psi_j(\boldsymbol{x})$ at the evaluation point \boldsymbol{x} are obtained by solving a constrained quadratic minimization problem. Thus, (3) is used as a constraint for the weighted norm minimization

$$\frac{1}{2} \sum_{j=1}^N \Psi_j^2(\boldsymbol{x}) w_j(\boldsymbol{x}) \rightarrow \min. \quad (4)$$

Here the w_j are positive (often radial) weight functions whose value increases with the distance from the center. The connection to Shepard's method (1) can be established if we identify $w_j(\boldsymbol{x}) = \frac{1}{\Phi_j(\boldsymbol{x})}$.

Using Lagrange multipliers $\lambda_j(\mathbf{x})$ (which will depend on the evaluation point \mathbf{x}) we can compute the values $\Psi_j(\mathbf{x})$ by solving a (small) linear system with Gram matrix $G(\mathbf{x})$ whose entries are given by

$$G_{k,\ell}(\mathbf{x}) = \sum_{j=1}^N p_k(\mathbf{x}_j) p_\ell(\mathbf{x}_j) \Phi_j(\mathbf{x}), \quad k, \ell = 1, \dots, m. \quad (5)$$

The resulting formula for the generating functions is

$$\Psi_j(\mathbf{x}) = \Phi_j(\mathbf{x}) \sum_{k=1}^m \lambda_k(\mathbf{x}) p_k(\mathbf{x}_j), \quad j = 1, \dots, N. \quad (6)$$

In the case $d = 0$, i.e., reproduction of constants, the matrix G consists of only a single element, $\sum_{j=1}^N \Phi_j(\mathbf{x})$. Therefore $\lambda_1(\mathbf{x}) = 1 / \sum_{j=1}^N \Phi_j(\mathbf{x})$, and the quasi-interpolant (2) together with the generating functions (6) is nothing but Shepard's method.

The approximation order $\mathcal{O}(h)$ mentioned earlier is with respect to a “mesh-size” h of the set of data sites. A common such measure is the so-called fill (or Hausdorff) distance $h = \sup_{\mathbf{x} \in \Omega} \min_{1 \leq j \leq N} \|\mathbf{x} - \mathbf{x}_j\|_2$, where $\Omega \subset \mathbb{R}^s$ is the computational domain under consideration. For general MLS approximation with reproduction of multivariate polynomials of degree d both Levin [6] and Wendland [12] proved approximation order $\mathcal{O}(h^{d+1})$.

If we define the discrete moments

$$\mu_\alpha = \sum_{j=1}^N (\mathbf{x}_j - \mathbf{x})^\alpha \Psi_j(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^s, \quad 0 \leq |\alpha| \leq d, \quad (7)$$

then solution of the Gram system (5) is equivalent to imposing the discrete moment conditions

$$\mu_\alpha = \delta_{\alpha,0}, \quad 0 \leq |\alpha| \leq d, \quad (8)$$

on the generating functions. Thus, following the proofs in [6] and [12], we can (theoretically) obtain quasi-interpolants of the form

$$\mathcal{Q}_h f(\mathbf{x}) = \sum_{j=1}^N f(\mathbf{x}_j) \Psi \left(\frac{\mathbf{x} - \mathbf{x}_j}{\rho_j} \right) \quad (9)$$

with approximation order $\mathcal{O}(h^{d+1})$. Here the generating functions $\Psi_j(\mathbf{x}) = \Psi(\mathbf{x} - \mathbf{x}_j)$ are assumed to be compactly supported with support size $\rho_j = ch$ ($c = \text{const.}$).

Equation (9) is an approximation scheme of the desired form (2). Indeed, for small values of s and d it is possible to analytically compute the generating functions by solving the moment equations (8). This was done in [1].

Since this computation is done as a preprocessing step, this method satisfies our matrix-free criterion stated in the introduction (but yields only limited approximation orders due to the limited range of d -values). Satisfying these moment conditions in arbitrary space dimensions, for higher approximation orders and for irregularly spaced data is a formidable task. In order to obtain practical quasi-interpolation schemes of this nature we have previously suggested making use of the concept of approximate approximation. This will lead to another matrix-free approximation scheme with the additional property of providing high approximation orders.

2.2 Approximate MLS Approximation for Regularly Spaced Centers

Maz'ya and Schmidt [8] formulated their approximate approximation method as a quasi-interpolation scheme in which continuous moment conditions of the form

$$\int_{\mathbb{R}^s} \mathbf{y}^\alpha \Psi(\mathbf{y}) d\mathbf{y} = \delta_{\alpha,0}, \quad 0 \leq |\alpha| \leq d, \quad (10)$$

are satisfied. Since they show that by satisfying (10) we also approximately satisfy the discrete moment conditions (8), a connection to MLS approximation can be established.

On a set of regularly spaced data points $\mathbf{x}_\nu = h\nu$ with spacing h Maz'ya and Schmidt suggest a quasi-interpolant of the form

$$\mathcal{M}_h f(\mathbf{x}) = \mathcal{D}^{-s/2} \sum_{\mathbf{x}_\nu \in B(\mathbf{x})} f(\mathbf{x}_\nu) \Psi \left(\frac{\mathbf{x} - \mathbf{x}_\nu}{\sqrt{\mathcal{D}}h} \right), \quad (11)$$

where $B(\mathbf{x})$ is a neighborhood of \mathbf{x} . In our numerical experiments we will take this to be the entire domain. Choosing a smaller ball while maintaining the approximation order given below in (12) depends on the decay rate of the generating function, and is closely related to fast evaluation methods.

The key ingredient in the approximation scheme (11) is the parameter \mathcal{D} which scales the generating function Ψ . It can be chosen to make a so-called *saturation error* so small that it does not affect numerical computations. The resulting approximation order estimate is then:

$$\|f - \mathcal{M}_h f\|_\infty = \mathcal{O}(h^{d+1} + \varepsilon_0(\Psi, \mathcal{D})) . \quad (12)$$

The saturation error ε_0 can be interpreted as the discrepancy between the continuous and discrete moment conditions, and its influence can be controlled by the choice of the parameter \mathcal{D} in (11). Since the analysis of this approximation method is based on Fourier techniques, the Fourier transform of the

generating function Ψ at the origin gives us an estimate for \mathcal{D} for any desired saturation error (see [8] or [2] for more details). The bottom line is that if \mathcal{D} is chosen large enough, then the saturation error will be smaller than the machine accuracy for any given computer, and therefore not noticeable in numerical computations. This means, that even though – theoretically – the quasi-interpolation scheme (11) fails to converge, it does converge for numerical purposes. Moreover, the rate of convergence, $d + 1$, can be arbitrarily high.

For radial generating functions we can use spherical coordinates to transform the multivariate moment conditions (10) to univariate conditions well known from the classical theory of orthogonal polynomials. In this special case the quasi-interpolant will be of the form

$$\mathcal{M}_h f(\mathbf{x}) = \mathcal{D}^{-s/2} \sum_{\mathbf{x}_\nu \in B(\mathbf{x})} f(\mathbf{x}_\nu) \psi \left(\frac{\|\mathbf{x} - \mathbf{x}_\nu\|^2}{\mathcal{D}h^2} \right), \quad (13)$$

where $\Psi(\mathbf{x}) = \psi(\|\mathbf{x}\|^2)$. The change from Ψ to ψ in the quasi-interpolant implies that the error estimate analogous to (12) will now be of the order $\mathcal{O}(h^{2d+2} + \varepsilon_0(\psi, \mathcal{D}))$.

Since, for radial functions, the moment conditions (10) are just univariate orthogonality conditions (for more details see [2]) it is straightforward to compute generating functions in a preprocessing step. A nice set of generating functions is related to the Gaussians. Table 1 lists these functions for space dimensions 1, 2, 3, and approximation orders $\mathcal{O}(h^2)$, $\mathcal{O}(h^4)$, and $\mathcal{O}(h^6)$.

Table 1

Generating functions ψ for approximate MLS approximation in \mathbb{R}^s .

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	$\frac{1}{\pi} e^{-\ \mathbf{x}\ ^2}$	$\frac{1}{\pi} (2 - \ \mathbf{x}\ ^2) e^{-\ \mathbf{x}\ ^2}$	$\frac{1}{\pi} \left(3 - 3\ \mathbf{x}\ ^2 + \frac{1}{2}\ \mathbf{x}\ ^4 \right) e^{-\ \mathbf{x}\ ^2}$
3	$\frac{1}{\pi^{3/2}} e^{-\ \mathbf{x}\ ^2}$	$\frac{1}{\pi^{3/2}} \left(\frac{5}{2} - \ \mathbf{x}\ ^2 \right) e^{-\ \mathbf{x}\ ^2}$	$\frac{1}{\pi^{3/2}} \left(\frac{35}{8} - \frac{7}{2}\ \mathbf{x}\ ^2 + \frac{1}{2}\ \mathbf{x}\ ^4 \right) e^{-\ \mathbf{x}\ ^2}$

A detailed analysis shows that the polynomial terms in Table 1 are given by generalized Laguerre polynomials with radial arguments [8], i.e., if $L_d^{s/2}$ is the generalized Laguerre polynomial of degree d , then

$$\Psi_\nu(\mathbf{x}) = \frac{1}{\pi^{s/2}} L_d^{s/2} \left(\frac{\|\mathbf{x} - \mathbf{x}_\nu\|^2}{\mathcal{D}h^2} \right) \exp \left(-\frac{\|\mathbf{x} - \mathbf{x}_\nu\|^2}{\mathcal{D}h^2} \right) \quad (14)$$

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

In summary, for regularly spaced centers, we can use the functions of Table 1

(or more generally (14)) in the quasi-interpolation formulation (13) to obtain an approximation method that satisfies both of our requirements to be matrix-free and have high (approximate) approximation order.

A method very similar to approximate approximation (called *corrected RKPM*) was also discussed in various papers by W.-K. Liu and co-workers (see [7] for a summary of those ideas). However, their work does not cover irregularly spaced data sites. The work of Maz'ya and Schmidt does, and we discuss this in the next section.

2.3 Irregularly Spaced Centers

If the irregularly spaced data sites do not differ too much from a regular reference grid, i.e., if there exists a smooth and nonsingular reparametrization $\gamma = (\gamma_1, \dots, \gamma_s)$ from \mathbb{R}^s to \mathbb{R}^s such that $\mathbf{x}_\nu = \gamma(h\nu)$, $\nu \in \mathbb{Z}^s$, then the quasi-interpolant (11) becomes

$$\mathcal{N}_h f(\mathbf{x}) = \mathcal{D}^{-s/2} \sum_{\mathbf{x}_\nu \in \gamma(B(\mathbf{x}))} f(\mathbf{x}_\nu) \Psi \left(\frac{\mathbf{x} - \mathbf{x}_\nu}{\sqrt{\mathcal{D}} h |\gamma'(h\nu)|^{1/s}} \right), \quad (15)$$

where $\gamma'(\mathbf{y})$ is the Jacobian of the reparametrization, i.e.,

$$\gamma'(\mathbf{y}) = \det \begin{pmatrix} \frac{\partial \gamma_1}{\partial y_1}(\mathbf{y}) & \dots & \frac{\partial \gamma_1}{\partial y_s}(\mathbf{y}) \\ \vdots & & \vdots \\ \frac{\partial \gamma_s}{\partial y_1}(\mathbf{y}) & \dots & \frac{\partial \gamma_s}{\partial y_s}(\mathbf{y}) \end{pmatrix}. \quad (16)$$

Maz'ya and Schmidt show in [8] that the quasi-interpolant (15) satisfies an error estimate of the same form as (12).

We see that (15) amounts to an individual scaling of the generating functions depending on the local deviation from a regular lattice.

In general, the map γ will not be known, and so we need to find an approximation J_ν to $|\gamma'(h\nu)|$ that maintains the desired approximation order. In particular, this means that we need (for radial generating functions)

$$|J_\nu - |\gamma'(h\nu)|| = \mathcal{O}(h^{2d+2}). \quad (17)$$

3 Approximating the Jacobian

The two methods presented here for obtaining higher-order approximations to the Jacobian (16) were essentially already suggested in [8]. We review them again below, and then present numerical experiments based on these approximations in the following section.

3.1 Finite Difference Approximations

An obvious means of approximating the Jacobian of the reparametrization γ is to replace the partial derivatives in (16) by finite difference approximations having the desired accuracy. We recall the following finite difference approximations to the first derivative of a given univariate function g with various accuracies:

$$\begin{aligned}
g'(x) &= \frac{g(x+h) - g(x-h)}{2h} + \mathcal{O}(h^2), \\
g'(x) &= 2 \frac{g(x+h) - g(x-h)}{3h} - \frac{g(x+2h) - g(x-2h)}{12h} + \mathcal{O}(h^4), \\
g'(x) &= 3 \frac{g(x+h) - g(x-h)}{4h} - 3 \frac{g(x+2h) - g(x-2h)}{20h} \\
&\quad + \frac{g(x+3h) - g(x-3h)}{60h} + \mathcal{O}(h^6). \tag{18}
\end{aligned}$$

These approximations are obtained by differentiating a polynomial of degree 2, 4, and 6, respectively, interpolating g at equally spaced points.

Since we need to approximate the Jacobian $\gamma'(h\nu)$ with entries $\frac{\partial \gamma_j}{\partial y_k}(h\nu)$, $j, k = 1, \dots, s$, and since $\gamma(h\nu) = \mathbf{x}_\nu$, the formulas (18) become

$$\begin{aligned}
\frac{\partial \gamma_j}{\partial y_k}(h\nu) &= \frac{x_{j,\nu+e_k} - x_{j,\nu-e_k}}{2h} + \mathcal{O}(h^2), \\
\frac{\partial \gamma_j}{\partial y_k}(h\nu) &= 2 \frac{x_{j,\nu+e_k} - x_{j,\nu-e_k}}{3h} - \frac{x_{j,\nu+2e_k} - x_{j,\nu-2e_k}}{12h} + \mathcal{O}(h^4), \\
\frac{\partial \gamma_j}{\partial y_k}(h\nu) &= 3 \frac{x_{j,\nu+e_k} - x_{j,\nu-e_k}}{4h} - 3 \frac{x_{j,\nu+2e_k} - x_{j,\nu-2e_k}}{20h} \\
&\quad + \frac{x_{j,\nu+3e_k} - x_{j,\nu-3e_k}}{60h} + \mathcal{O}(h^6). \tag{19}
\end{aligned}$$

where $x_{j,\nu}$ denotes the j -th component of \mathbf{x}_ν , and \mathbf{e}_k is the k -th standard unit vector in \mathbb{R}^s .

By replacing all partial derivatives in (16) with finite difference approximations

of the type (19) we can get an approximation of the Jacobian for the quasi-interpolant (15) with the desired accuracy.

In the Section 4.1 we will present a strategy for identifying the points $\mathbf{x}_{\nu \pm i\mathbf{e}_k}$, $i = 1, 2, 3$, and $k = 1, \dots, s$.

3.2 Approximate Areas

Another way of adjusting the scale of the generating function centered at \mathbf{x}_ν in our quasi-interpolant is to base the correction factor on some measure of area (or volume in higher space dimensions) defined by points in the neighborhood of \mathbf{x}_ν . However, using simple areas will only provide an approximation of order $\mathcal{O}(h^2)$ useful for working with the basic Gaussian generating functions (see below). If we want to take advantage of the more accurate generating functions listed in Table 1 then we need to devise higher-order approximations to the Jacobian of the reparametrization as specified in (17).

To this end we let $M = 2d + 2$, and consider a cube Q containing the origin (in the regular reference lattice) with area (volume)

$$|Q| = \int_Q d\mathbf{x}.$$

Then $\gamma(Q)$ denotes the transformed cube which has area

$$|\gamma(Q)| = \int_{\gamma(Q)} d\mathbf{x} = \int_Q |\gamma'(\mathbf{y})| d\mathbf{y}.$$

Moreover, if Q_ν denotes a shifted and scaled copy of Q , i.e., $Q_\nu = h\nu + hQ$, then

$$|\gamma(Q_\nu)| = \int_{\gamma(Q_\nu)} d\mathbf{x} = \int_{Q_\nu} |\gamma'(\mathbf{y})| d\mathbf{y} = h^s \int_Q |\gamma'(\mathbf{y})| d\mathbf{y}. \quad (20)$$

Performing a Taylor expansion of the Jacobian coupled with integration over the cube Q yields

$$\int_Q |\gamma'(\mathbf{y})| d\mathbf{y} = \sum_{|\alpha|=0}^{M-1} \frac{D^\alpha(|\gamma'(h\nu)|)}{\alpha!} h^{|\alpha|} \int_Q \mathbf{y}^\alpha d\mathbf{y} + \mathcal{O}(h^M)$$

or

$$\int_Q |\gamma'(\mathbf{y})| d\mathbf{y} = |Q| |\gamma'(h\nu)| + \sum_{|\alpha|=1}^{M-1} \frac{D^\alpha(|\gamma'(h\nu)|)}{\alpha!} h^{|\alpha|} \int_Q \mathbf{y}^\alpha d\mathbf{y} + \mathcal{O}(h^M). \quad (21)$$

Using (20) we see that the left-hand side

$$\int_Q |\gamma'(\mathbf{y})| d\mathbf{y} = \frac{|\gamma(Q_\nu)|}{h^s}. \quad (22)$$

If we choose the cube $Q^1 = [-1, 1]^s$, then

$$\int_{Q^1} \mathbf{y} d\mathbf{y} = 0, \quad (23)$$

and (21), (22) and (23) imply that we can approximate $|\gamma'(h\nu)|$ by

$$J_\nu = \frac{|\gamma(Q_\nu^1)|}{h^s |Q^1|} = \frac{|\gamma(Q_\nu^1)|}{(2h)^s} \quad (24)$$

with accuracy $\mathcal{O}(h^2)$.

In order to obtain an $\mathcal{O}(h^4)$ approximation to the Jacobian we need to consider the next term in (21). Using the cube Q^1 we get

$$\int_{Q^1} |\gamma'(\mathbf{y})| d\mathbf{y} = 2^s |\gamma'(h\nu)| + \frac{2^s h^2}{6} \sum_{j=1}^s D^{2e_j}(|\gamma'(h\nu)|) + \mathcal{O}(h^4) \quad (25)$$

since, for $|\alpha| = 2$,

$$\int_{Q^1} \mathbf{y}^{e_j + e_k} d\mathbf{y} = \begin{cases} 0, & j \neq k \in \{1, \dots, s\}, \\ \frac{2^s}{3}, & j = k, \end{cases}$$

and, for $|\alpha| = 3$,

$$\int_{Q^1} \mathbf{y}^\alpha d\mathbf{y} = 0.$$

In order to remove the terms of order $\mathcal{O}(h^2)$ in (25) we need to use another expansion of this type coming from a different cube Q^2 .

For example, we can simply choose Q^2 as a scaled version of Q^1 , e.g., $Q^2 = [-2, 2]^s$. Then we get

$$\int_{Q^2} |\gamma'(\mathbf{y})| d\mathbf{y} = 4^s |\gamma'(h\nu)| + \frac{4^{s+1} h^2}{6} \sum_{j=1}^s D^{2e_j}(|\gamma'(h\nu)|) + \mathcal{O}(h^4) \quad (26)$$

which – by subtracting (26) from 2^{s+2} times (25) – leads to the following approximation to the Jacobian valid in any space dimension:

$$J_\nu = \frac{4|\gamma(Q_\nu^1)| - |\gamma(Q_\nu^2)|/2^s}{3(2h)^s} + \mathcal{O}(h^4). \quad (27)$$

For approximation in \mathbb{R}^2 this reads

$$J_\nu = \frac{|\gamma(Q_\nu^1)| - |\gamma(Q_\nu^2)|/16}{3h^2} + \mathcal{O}(h^4). \quad (28)$$

Even higher-order approximations can be achieved by using several different cubes, and forming linear combinations that eliminate the desired terms in (21).

If we restrict ourselves to the case $s = 2$ and take Q^3 with vertices at $\pm \mathbf{e}_j$, $j = 1, 2$, then we have

$$\int_{Q^3} |\gamma'(\mathbf{y})| d\mathbf{y} = 2|\gamma'(h\boldsymbol{\nu})| + \frac{h^2}{6} \left(D^{(2,0)}(|\gamma'(h\boldsymbol{\nu})|) + D^{(0,2)}(|\gamma'(h\boldsymbol{\nu})|) \right) + \mathcal{O}(h^4). \quad (29)$$

Therefore, for $s = 2$ we can subtract (25) from 4 times (29) to get an $\mathcal{O}(h^4)$ approximation of the Jacobian of the form

$$J_{\boldsymbol{\nu}} = \frac{|\gamma(Q_{\boldsymbol{\nu}}^3)| - |\gamma(Q_{\boldsymbol{\nu}}^1)|/4}{h^2}. \quad (30)$$

However, choosing the vertices in this way does not generalize naturally to higher dimensions.

We now present and implement a strategy for identifying the vertices of the cubes from the set of irregularly spaced centers.

4 Numerical Experiments

4.1 Implementation of Approximate Jacobians

The methods for approximating the Jacobian described in the previous section are based on the assumption that the locations of the irregularly spaced data sites are obtained by a smooth nonsingular transformation from a regular reference grid. Therefore we can use neighboring information in the reference grid to select the corresponding irregularly spaced points needed for the approximate formulas.

In the numerical experiments below we have used the $\mathcal{O}(h^2)$ finite difference approximation listed in (19) as well as the $\mathcal{O}(h^4)$ area approximation (30). Other approximations have also been implemented and tested.

If $s = 2$ then the basic symmetric finite difference formula with $\mathcal{O}(h^2)$ accuracy employs the direct neighbors of the center $h\boldsymbol{\nu}$ to the left, right, top and bottom. Therefore, in the set of irregularly spaced points we select the four nearest neighbors of $\mathbf{x}_{\boldsymbol{\nu}} = \gamma(h\boldsymbol{\nu})$ and sort them according to their relative positions. Then the approximate Jacobian is computed based on these points.

Similarly, for the $\mathcal{O}(h^4)$ approximation we require the eight nearest neighbors

of \mathbf{x}_ν . They are then divided into two groups. The four nearest neighbors forming the cube Q_ν^3 (corresponding to a diamond-like shape in the reference grid), and the other four points yielding the vertices of the cube Q_ν^1 (corresponding to a square shape in the reference grid).

We use the ANN (approximate nearest neighbor) library [9] to compute the required lists of nearest neighbors for each center \mathbf{x}_ν .

4.2 Equally Spaced Centers

We use the following mollified version g of “Franke’s function” f given by

$$\begin{aligned} f(x, y) &= \frac{3}{4} \left[\exp \left(-\frac{(9x-2)^2}{4} - \frac{(9y-2)^2}{4} \right) + \exp \left(-\frac{(9x+1)^2}{49} - \frac{(9y+1)^2}{10} \right) \right] \\ &\quad + \frac{1}{2} \exp \left(-\frac{(9x-7)^2}{4} - (9y-3)^2 \right) - \frac{1}{5} \exp \left(-(9x-4)^2 - (9y-7)^2 \right), \\ g(x, y) &= 15 \exp \left(\frac{-1}{1-4(x-1/2)^2} \right) \exp \left(\frac{-1}{1-4(y-1/2)^2} \right) f(x, y), \end{aligned}$$

where $(x, y) \in [0, 1] \times [0, 1]$. In order for the quasi-interpolation schemes to perform well it is necessary that the data be mollified around the boundary.

Since the approximation orders for approximate approximation are maximum norm errors (cf. (12)) we use the corresponding discrete errors to document our numerical experiments. The ℓ_∞ errors reported in Tables 2–5 are always computed on one more refinement of the finest mesh, i.e., a mesh \mathcal{G} of 257×257 equally spaced points in $[0, 1] \times [0, 1]$. Specifically,

$$\ell_\infty\text{-error} = \max_{\mathbf{x}_j \in \mathcal{G}} |g(\mathbf{x}_j) - \mathcal{M}_h g(\mathbf{x}_j)|, \quad ,$$

where $\mathcal{M}_h g$ (or $\mathcal{N}_h g$) is the approximation to g obtained by the various approximation schemes. For the examples reported below using irregular data sites the error is also computed on the fine regular mesh \mathcal{G} .

In Table 2 we list – as benchmarks – the results of computations on a series of regular grids. The generating functions are the Gaussian, “linear” and “quadratic” Laguerre generating functions listed in the second row of Table 1. The “rate” columns indicate the numerical approximation order achieved by each method. In all three cases we chose the parameter $\mathcal{D} = 3$ in (13).

The convergence behavior shown in Table 1 is typical for approximate approximation. On a coarse grid the smoothing effect of selecting a fixed parameter

Table 2
Approximate MLS approximation with regular centers in \mathbb{R}^2 .

Mesh	Gaussian		“linear” Laguerre		“quadratic” Laguerre	
	ℓ_∞ -error	rate	ℓ_∞ -error	rate	ℓ_∞ -error	rate
3×3	1.023		1.004		9.887e-01	
5×5	8.348e-01	0.294	6.808e-01	0.562	5.603e-01	0.819
9×9	5.903e-01	0.500	3.651e-01	0.899	2.287e-01	1.293
17×17	2.691e-01	1.133	7.440e-01	2.295	3.955e-02	2.532
33×33	8.206e-02	1.714	1.714e-02	2.118	6.943e-03	2.510
65×65	2.162e-02	1.924	2.624e-03	2.708	7.809e-04	3.152
129×129	5.479e-03	1.981	2.743e-04	3.258	6.353e-05	3.620

\mathcal{D} for all mesh sizes is clearly visible, and the corresponding approximation order is low. By refining the grid, the smoothing effect is less dramatic, and eventually the predicted approximation order is reached. Finally, if we would be able to refine the mesh even further so that the errors approach the saturation error ε_0 (cf. (12)) associated with our choice of \mathcal{D} , then convergence would stall. Note that for the “quadratic” Laguerre generating function we do not achieve the predicted order $\mathcal{O}(h^6)$. Most likely this is due to smoothing effects induced by the choice of \mathcal{D} , and the convergence order would show up in later iterations on a finer mesh.

4.3 Irregularly Spaced Centers

We use two different sets of irregularly distributed points. The “smooth” point set is obtained from a regular grid via the mapping

$$\begin{aligned} x &= (19\xi + \eta^2)/20 \\ y &= (\xi^2 + 49\eta^2)/50, \end{aligned}$$

where ξ and η are points on a regular grid in $[0, 1] \times [0, 1]$. This point set satisfies the hypotheses of the theory outlined in Section 3. The second set of points are the so-called *Halton* points. They are defined as follows. Consider the p -adic expansion of the nonnegative integer n , i.e.,

$$n = \sum_j n_j p^j$$

with some prime base p . If we define a function H by

$$H(n, p) = \sum_j n_j p^{-j-1},$$

then the two-dimensional Halton points are given by

$$\mathbf{x}_{i+1} = [H(i, p_x), H(i, p_y)], \quad i = 0, 1, \dots, N - 1,$$

where p_x and p_y are two different prime numbers. Our Halton points were generated with $p_x = 2$ and $p_y = 3$. The Halton points are randomly distributed in the unit square, and no smooth mapping from a regular grid exists. More details on Halton points can be found, e.g., in [10].

The two different types of data sets are displayed in Figure 1 for $N = 289$.

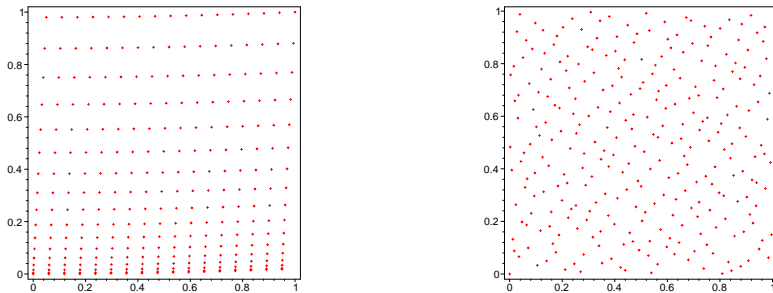


Fig. 1. 289 “smooth” points (left) and 289 Halton points (right) in $[0, 1] \times [0, 1]$.

We present three different sets of experiments. First, in Table 3 we perform the standard quasi-interpolation scheme (13), i.e., without using any scale adjustment for the Gaussian generating functions. As expected, on both sets of irregularly spaced data, the method fails to converge properly. For this experiment we used $\mathcal{D} = 5$.

Next, we test the performance of one of the finite difference approximations for the Jacobian. In Table 4 we used the $\mathcal{O}(h^2)$ approximation listed in (19). The approximation at each point \mathbf{x}_ν is computed based on the four nearest neighbors of \mathbf{x}_ν , as explained earlier. Again, we use $\mathcal{D} = 5$ and Gaussian generating functions.

Figure 2 shows the approximation obtained on the two different sets of 1089 points. The graph of the approximation computed over the “smooth” points is very close to that of the mollified “Franke function” g . In general, the approximations computed over the Halton points are “rougher” than those computed over the points obtained by a smooth mapping from a regular mesh. This is in accordance with the assumptions made in Section 2.3.

Finally, in Table 5 we present the results of computations based on “linear” Laguerre generating functions $\psi(\mathbf{x}) = \frac{1}{\pi} (2 - \|\mathbf{x}\|^2) e^{-\|\mathbf{x}\|^2}$ with an added $\mathcal{O}(h^4)$

Table 3

Approximate MLS approximation with Gaussians using “smooth” points and Halton points as centers in \mathbb{R}^2 ; no correction added.

Mesh	“smooth” points		Halton points	
	ℓ_∞ -error	rate	ℓ_∞ -error	rate
9	1.009		9.382e-01	
25	9.051e-01	0.156	8.071e-01	0.217
81	6.866e-01	0.399	5.680e-01	0.507
289	3.533e-01	0.958	2.533e-01	1.165
1089	1.734e-01	1.027	9.082e-02	1.480
4225	1.982e-01	-0.193	8.755e-02	0.053
16641	2.194e-01	-0.147	8.219e-02	0.091

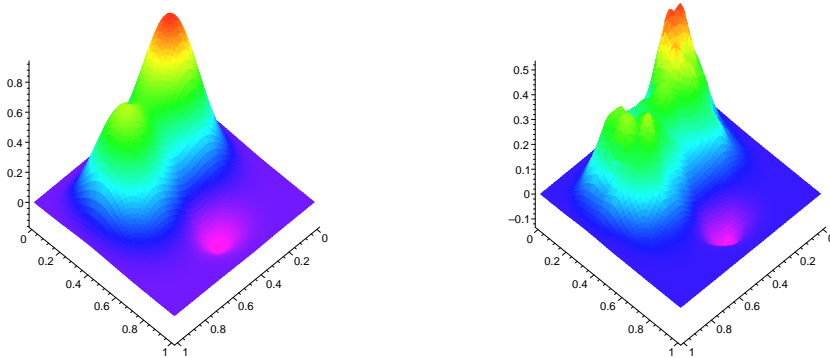


Fig. 2. Quasi-interpolant to mollified “Franke function” g using 1089 “smooth” points (left) and 1089 Halton points (right) in $[0, 1] \times [0, 1]$.

correction based on the area approximation (30) of the Jacobian. Again, we compare what happens for “smooth” points and for Halton points. This time we begin the experiments with a set of 25 points since the method used to approximate the Jacobian uses the 8 nearest neighbors of each center. Also, in this series of experiments we let $\mathcal{D} = 10$. The results are similar to the previous experiment. However, the rate of convergence is not as good. In fact, saturation seems to play a role during the later stages of the experiment.

Table 4

Approximate MLS approximation with Gaussians using “smooth” points and Halton as centers in \mathbb{R}^2 ; $\mathcal{O}(h^2)$ finite difference correction applied.

Mesh	“smooth” points		Halton points	
	ℓ_∞ -error	rate	ℓ_∞ -error	rate
9	1.011		9.557e-01	
25	9.259e-01	0.126	8.758e-01	0.126
81	6.965e-01	0.411	7.078e-01	0.307
289	3.753e-01	0.892	5.809e-01	0.285
1089	1.302e-01	1.527	6.683e-01	-0.202
4225	4.320e-02	1.592	6.079e-01	0.137
16641	1.216e-02	1.828	6.699e-01	-0.140

Table 5

Approximate MLS approximation with “linear” Laguerre generating functions using “smooth” points and Halton as centers in \mathbb{R}^2 ; $\mathcal{O}(h^4)$ area correction applied.

Mesh	“smooth” points		Halton points	
	ℓ_∞ -error	rate	ℓ_∞ -error	rate
25	9.010e-01		8.577e-01	
81	6.187e-01	0.542	6.561e-01	0.387
289	2.510e-01	1.301	2.909e-01	1.174
1089	9.183e-02	1.451	3.855e-01	-0.407
4225	7.057e-02	0.380	2.719e-01	0.504
16641	5.189e-02	0.444	3.459e-01	-0.347

5 Concluding Remarks

As the experiments of Table 3 show, it is necessary to apply a local correction factor to the scale of the generating functions when applying quasi-interpolation formulas on irregularly spaced points. However, as the experiments with the truly scattered Halton points show, the method used in this paper to adjust the scale relies on the fact that the centers are a smooth reparametrization of a regular grid, i.e., the method does not work in a reasonable way for the Halton points. Even for the points obtained with a smooth mapping the scheme for irregularly spaced points does not compare very favorably with the one for regularly spaced data (cf. Tables 2, 4 and 5).

Overall, for regular data we recommend the use of “linear” Laguerre functions, whereas for irregularly spaced data it does – at least so far – not seem to be beneficial to use those functions instead of regular Gaussians. Further work is needed to obtain a robust adjustment mechanism for the scale of the generating functions for high-order matrix-free quasi-interpolants such as (2) or (13).

References

- [1] G. E. Fasshauer, Matrix-free multilevel moving least-squares methods, in *Approximation Theory X: Wavelets, Splines, and Applications*, C. K. Chui, L. L. Schumaker, and J. Stöckler (eds.), Vanderbilt University Press, 2002, 271–281.
- [2] G. E. Fasshauer, 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.
- [3] G. E. Fasshauer, Approximate moving least-squares approximation for time-dependent PDEs, in *WCCM V, Fifth World Congress on Computational Mechanics* (<http://wccm.tuwien.ac.at>), H. A. Mang, F. G. Rammerstorfer, and J. Eberhardsteiner (eds.), Vienna University of Technology, 2002.
- [4] G. E. Fasshauer, Approximate moving least-squares approximation: a fast and accurate multivariate approximation method, in *Curve and Surface Fitting: Saint-Malo 2002*, A. Cohen, J.-L. Merrien, and L. L. Schumaker (eds.), Nashboro Press, 2003, 139–148.
- [5] P. Lancaster, K. Šalkauskas, Surfaces generated by moving least squares methods, *Math. Comp.* (37) 1981, 141–158.
- [6] D. Levin, The approximation power of moving least-squares, *Math. Comp.* (67) 1998, 1517–1531.
- [7] S. Li, W. K. Liu, Meshfree and particle methods and their applications, *Applied Mechanics Review* (55) 2002, 1–34.
- [8] V. Maz’ya, G. Schmidt, On quasi-interpolation with non-uniformly distributed centers on domains and manifolds, *J. Approx. Theory* (110) 2001, 125–145.
- [9] D. M. Mount, S. Arya, ANN: Library for Approximate Nearest Neighbor Searching, <http://www.cs.umd.edu/~mount/ANN/>.
- [10] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*, CBMS-NSF Regional Conference Series in Applied Mathematics, 63, SIAM, Philadelphia, PA, 1992.

- [11] D. Shepard, A two dimensional interpolation function for irregularly spaced data, Proc. 23rd Nat. Conf. ACM, 1968, 517–523.
- [12] H. Wendland, Local polynomial reproduction and moving least squares approximation, IMA J. Numer. Anal. (21) 2001, 285–300.