A rational interpolation scheme with super-polynomial rate of convergence

By Q. Wang, P. Moin and G. Iaccarino

1. Motivation and objectives

The purpose of this study is to construct a high-order interpolation and scheme for arbitrary scattered datasets. The resulting function approximation is an interpolation function when the dataset is exact, or a regression if measurement errors are present. We represent each datapoint with a Taylor series, and the approximation error as a combination of the derivatives of the the target function. A weighted sum of the square of the coefficient of each derivative term in the approximation error is minimized to obtain the interpolation approximation. The resulting approximation function is a high-order rational function with no poles. When measurement errors are absent, the interpolation approximation converges to the target function faster than any polynomial rate of convergence.

2. Background

Let $\hat{f}(x_i) \approx f(x_i)$ be measurements of the target function f at $x_i, i = 1, \ldots, n$. The measurement errors $\hat{f}(x_i) - f(x_i)$ are mutually independent random variables with zero mean and standard deviation σ_i. We assume that at most one measurement is given at each point unless the measurements are inexact, i.e., if $x_i = x_j$ for $i \neq j$, then both $\sigma_i > 0$ and $\sigma_j > 0$. This is called the non-redundancy condition. We construct an approximation function $\tilde{f}$ based on these measurements. Specifically, the value of $\tilde{f}$ at any point x is constructed as

$$\tilde{f}(x) = \sum_{i=1}^{n} a_i \hat{f}(x_i), \tag{2.1}$$

where a_i are functions of x and satisfy

$$\sum_{i=1}^{n} a_i \equiv 1. \tag{2.2}$$

Under this normalization condition, we choose a_i for each x, so that the approximation error at this point $\tilde{f}(x) - f(x)$ is small. Specifically, by expanding each $f(x_i)$ using Taylor's theorem

$$f(x_i) = f(x) + \sum_{k=1}^{N} f^{(k)}(x) \frac{(x_i - x)^k}{k!} + f^{(N+1)}(\xi_i) \frac{(x_i - x)^{N+1}}{(N+1)!},$$

the approximation error becomes

$$\begin{aligned}\tilde f(x) - f(x) = &\sum_{k=1}^{N} f^{(k)}(x)\left(\sum_{i=1}^{n} a_i \frac{(x_i - x)^k}{k!}\right)\\ &+ \sum_{i=1}^{n} f^{(N+1)}(\xi_i)\left(a_i \frac{(x_i - x)^{N+1}}{(N+1)!}\right)\\ &+ \sum_{i=1}^{n}\left(\hat f(x_i) - f(x_i)\right) a_i \ ,\end{aligned} \tag{2.3}$$

where each ξ_i lies between x and x_i. In this formula of the approximation error, both the derivatives of the target function $f^{(k)}(x)$, $f^{(N+1)}(\xi_i)$ and the measurement errors $\hat f(x_i) - f(x_i)$ are unknown. The approximation error $\tilde f(x) - f(x)$ is a linear function of these unknowns. In order to make the approximation error small, we choose a_i to minimize a specific norm of this linear function, which is a weighted sum of the square of the coefficients of these unknowns

$$\begin{aligned}\mathcal{Q}(x, a_1, \ldots, a_n) = &\sum_{k=1}^{N} w_k^2 \left(\sum_{i=1}^{n} a_i \frac{(x_i - x)^k}{k!}\right)^2\\ &+ \sum_{i=1}^{n} w_{N+1}^2 \left(a_i \frac{(x_i - x)^{N+1}}{(N+1)!}\right)^2\\ &+ \sum_{i=1}^{n} \sigma_i^2 a_i^2 \ ,\end{aligned} \tag{2.4}$$

where the weight on the coefficient of the measurement error $\hat f(x_i) - f(x_i)$ is the variance of the measurement error σ_i^2, and the weights on the coefficients of the derivatives of f are an infinite series of positive input parameters $w_k > 0, k = 1, 2, \ldots$. For best approximation results, these parameters should be chosen to reflect the magnitude of $f^{(k)}$, i.e.,

$$w_k \approx \|f^{(k)}\|, \quad k = 1, 2, \ldots$$

The influence of these parameters on the approximation error is illustrated in Sec. 5, and a proper choice of w_k is discussed in Sec. 6. With $\mathcal{Q}$ as the objective of minimization and the normalization constraint (2.2), the interpolation coefficients a_i at x are the solution of the constraint minimization problem

$$\min \mathcal{Q}(x, a_1, \ldots, a_n) \quad \text{s.t.} \quad \sum_{i=1}^{n} a_i = 1 \ . \tag{2.5}$$

The existence and uniqueness of the solution is proven in Sec. 3. With a set of a_i determined at each x, the approximation function $\tilde f$ can then be calculated by Eq. (2.1).

3. Existence and uniqueness

This section proves the existence and uniqueness of the solution of Eq. (2.5), so that the approximation $\tilde f$ based on Eqs. (2.1) and (2.5) is well defined. Let $a = [a_1 \ldots a_n]$, the objective of minimization is a quadratic function of a, and can be written in matrix form as

$$\mathcal{Q}(x, a_1, \ldots, a_n) = a\, A\, a^{\mathrm{T}}, \tag{3.1}$$

where A is an n by n symmetric matrix

$$A_{ij} = \begin{cases} \sum_{k=1}^{N} \left(\frac{w_k}{k!}\right)^2 (x_i - x)^k (x_j - x)^k & i \neq j \\ \sum_{k=1}^{N+1} \left(\frac{w_k}{k!}\right)^2 (x_i - x)^{2k} + \sigma_i^2 & i = j \end{cases} \tag{3.2}$$

The matrix form of the constraint quadratic programming is

$$\min \; a\,A\,a^{\mathrm{T}} \quad \text{s.t.} \quad e\,a^{\mathrm{T}} = 1 \;, \tag{3.3}$$

where $e = [1\ldots,1]$ has the same length as a. In order to prove the existence and uniqueness of its solution, we first prove some properties of the matrix A.

LEMMA 1. *Let $w_k > 0$ for all $k > 0$; x_i and σ_i satisfy the non-redundancy condition, $x_i = x_j$ and $i \neq j \Longrightarrow \sigma_i > 0$ and $\sigma_j > 0$. Then A is positive definite if $x \neq x_i$ for any i such that $\sigma_i = 0$.*

Proof. To prove that A is positive definite, we only need to prove that $a\,A\,a^{\mathrm{T}} = \mathcal{Q}(x, a_1, \ldots, a_n) > 0$ whenever $a_j \neq 0$ for some j. We prove this in two cases: (1) $x \neq x_i$ for any i; and (2) $x = x_i$ but $\sigma_i > 0$. In Case (1), because $x \neq x_j$,

$$\mathcal{Q}(x, a_1, \ldots, a_n) \geq w_{N+1}^2 \left(a_j \frac{(x_j - x)^{N+1}}{(N+1)!} \right)^2 > 0 \;. \tag{3.4}$$

In Case (2), if $x_i = x_j$, from the non-redundancy condition, $\sigma_j > 0$, and

$$\mathcal{Q}(x, a_1, \ldots, a_n) \geq \sigma_j^2 a_j^2 > 0 \;.$$

On the other hand, if $x = x_i \neq x_j$, then (3.4) applies. In all these possible cases, $a\,A\,a^{\mathrm{T}} = \mathcal{Q}(x, a_1, \ldots, a_n) > 0$ whenever $a \neq 0$. Therefore, A is positive definite. □

LEMMA 2. *Let x_i and σ_i satisfy the non-redundancy condition as stated in Lemma 1. If $x = x_i$ and $\sigma_i = 0$, then A is positive semi-definite and rank one deficient. The nullspace of A is $\{a \mid a_j = 0, \forall j \neq i\}$.*

Proof. Because $a\,A\,a^{\mathrm{T}} = \mathcal{Q}(x, a_1, \ldots, a_n) \geq 0$ by definition, A is positive semi-definite. In addition, if $a_j = 0$ for all $j \neq i$, then

$$\mathcal{Q}(x, a_1, \ldots, a_n) = \sum_{k=1}^{N} w_k^2 \left(a_i \frac{(x_i - x)^k}{k!} \right)^2 + w_{N+1}^2 \left(a_i \frac{(x_i - x)^{N+1}}{(N+1)!} \right)^2 + \sigma_i^2 a_i^2 \;.$$

Because $x = x_i$ and $\sigma_i = 0$, all terms in the formula above are 0, and $a\,A\,a^{\mathrm{T}} = \mathcal{Q}(x, a_1, \ldots, a_n) = 0$, and the nullspace of A includes $\{a \mid a_j = 0, \forall j \neq i\}$. In order to prove that A is rank one deficient and $\{a \mid a_j = 0, \forall j \neq i\}$ is the nullspace, we need to prove that any $a\,A\,a^{\mathrm{T}} > 0$ for any a with $a_j \neq 0$, $j \neq i$. In fact, because $\sigma_i = 0$ and the non-redundancy condition, $x = x_i \neq x_j$ if $i \neq j$. Therefore, (3.4) applies if $a_j \neq 0$ for $j \neq i$. In other words, $a\,A\,a^{\mathrm{T}} > 0$ for any a is not in the one dimensional linear subspace $\{a \mid a_j = 0, \forall j \neq i\}$. Therefore, A is rank one deficient with nullspace $\{a \mid a_j = 0, \forall j \neq i\}$. □

Lemma 1 and Lemma 2 show that the matrix A is always positive semi-definite. In addition, it is rank one deficient when x coincides with a node with no measurement error ($x = x_i$ and $\sigma_i = 0$), and full rank otherwise. With this property of A, we can prove that the constraint quadratic programming (3.1) has a unique solution for any x.

THEOREM 1. *If x_i and σ_i satisfy the non-redundancy condition as stated in Lemma 1, then the solution of the quadratic programming (3.1) exists and is unique.*

Proof. Since A is positive semi-definite, the existence of a solution follows the two trivial facts: there exists a that satisfies the constraint; and $a\,A\,a^{\mathrm{T}}$ is bounded from below on the feasible region. To prove its uniqueness, we use the fact that a necessary condition for an optimal solution of the constraint quadratic programming is that it must satisfy the Karush-Kuhn-Tucker condition Nocedal & Wright (2000)

$$\begin{bmatrix} 2A & e^{\mathrm{T}} \\ e & 0 \end{bmatrix} \begin{bmatrix} a^{\mathrm{T}} \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} . \tag{3.5}$$

To prove the uniqueness of the solution, we only need to prove that the matrix in the above linear system is full rank. In other words, it is sufficient to prove that $a \neq 0$ or $\lambda \neq 0 \Longrightarrow 2A\,a^{\mathrm{T}} + \lambda\,e^{\mathrm{T}} \neq 0$ or $e\,a^{\mathrm{T}} \neq 0$. We first look at the case when $a \neq 0$. In this case, if $e\,a^{\mathrm{T}} = 0$, then $a \notin \{a \mid a_j = 0, \forall j \neq i\}$ for any i. By Lemma 1 and Lemma 2, a is not in the nullspace of A even if A is singular, and we have $a\,(2A\,a^{\mathrm{T}} + \lambda\,e^{\mathrm{T}}) = 2\,a\,A\,a^{\mathrm{T}} > 0$. Therefore, either $e\,a^{\mathrm{T}} \neq 0$ or $2A\,a^{\mathrm{T}} + \lambda\,e^{\mathrm{T}} \neq 0$ when $a \neq 0$. The second case is when $a = 0$ but $\lambda \neq 0$. In this case, $2A\,a^{\mathrm{T}} + \lambda\,e^{\mathrm{T}} = \lambda\,e^{\mathrm{T}} \neq 0$. Therefore, the matrix

$$\begin{bmatrix} 2A & e^{\mathrm{T}} \\ e & 0 \end{bmatrix}$$

is full rank, and the solution of the K-K-T system (3.5) is unique. Thus the solution of the quadratic programming (3.1) exists and is unique. □

4. Rational form

This section proves that our interpolation or regression approximation function $\tilde{f}$ is a rational function with no poles in the real line $(-\infty, +\infty)$, and therefore is bounded, continuous and infinitely differentiable. We first prove that it is a rational function.

THEOREM 2. *The approximation function $\tilde{f}$ given by Eqs. (2.1) and (2.5) is a rational function of x that has no pole in $(-\infty, +\infty)$.*

Proof. From Eq. (3.5), we have

$$\begin{bmatrix} A & e^{\mathrm{T}} \\ e & 0 \end{bmatrix} \begin{bmatrix} a^{\mathrm{T}} \\ \frac{\lambda}{2} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} .$$

Therefore,

$$\begin{bmatrix} a^{\mathrm{T}} \\ \frac{\lambda}{2} \end{bmatrix} = \begin{bmatrix} A & e^{\mathrm{T}} \\ e & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{\begin{bmatrix} A & e^{\mathrm{T}} \\ e & 0 \end{bmatrix}^{*}}{\begin{vmatrix} A & e^{\mathrm{T}} \\ e & 0 \end{vmatrix}} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{4.1}$$

where $[\,\cdot\,]^*$ denotes the adjugate matrix, and $|\cdot|$ denotes the determinant. But according to Eq. (3.2), each element of the matrix A is a polynomial of x. Therefore, both A^*_{ij} and all elements of $\begin{bmatrix} A & e^{\mathrm{T}} \\ e & 0 \end{bmatrix}^*$ are polynomials of x. As a result, the r.h.s. of Eq. (4.1) is a rational function of x. So the vector a is a rational function of x, and $\tilde{f} = \sum_{i=1}^n a_i \tilde{f}(x_i)$ is a rational function of x. This proves the first part of Theorem 2.

In order to prove that the rational function has no poles in the real line, it is sufficient to prove that the denominator $\begin{vmatrix} A & e^{\mathrm{T}} \\ e & 0 \end{vmatrix}$ has no zeros in $(-\infty, \infty)$. In fact, recall that $e = [1, \dots, 1]$, from the definition of matrix determinant and cofactors, we have

$$\begin{vmatrix} A & e^{\mathrm{T}} \\ e & 0 \end{vmatrix} = \sum_{i,j=1}^n A_{ij}^* \ .$$

We show that it is non-zero in two cases. First, when $x = x_i$ and $\sigma_i = 0$ for some i, the ith column and row of matrix A are zero according to Eq. (3.2). Therefore, all cofactors other than A_{ii}^* are zero, and the denominator simply equals A_{ii}^*, which is non-zero because A is only one rank deficient (Lemma 1). In the other case, x is not equal to any x_i, and A is positive definite (Lemma 1). Therefore, A^* is also positive definite, and the denominator is

$$\sum_{i,j=1}^n A_{ij}^* = eA^*e^{\mathrm{T}} > 0 \ .$$

The denominator of the rational function $\tilde{f}$ is therefore non-zero for any $x \in \mathrm{R}$. Hence $\tilde{f}$ has no poles in $(-\infty, +\infty)$. □

The following property of our interpolation and regression approximation function derives naturally from Theorem 2.

Corollary 1. *The interpolant $\tilde{f}$ given by Eqs. (2.1) and (2.5) is continuous and infinitely differentiable.*

5. Interpolation approximation

This section discusses the case when the measurement errors are 0, i.e., $f(x_i) = \hat{f}(x_i)$ and $\sigma_i = 0$ for all $i = 1, \dots, n$. We first prove that our approximation function $\tilde{f}(x)$ defined by Eqs. (2.1) and (2.5) is indeed an interpolation, in that the approximation function goes through each datapoint $(x_i, f(x_i))$.

Theorem 3. *If $\sigma_i = 0$, then $\tilde{f}(x_i) = f(x_i)$.*

Proof. Let $x = x_i$, then from Eq. (3.2), the ith row and column of the matrix A are all 0. Therefore, the positive semi-definite quadratic form Eq. (3.1) is equal to 0 when a is equal to

$$a_j = \begin{cases} 1 & j = i; \\ 0 & j \leq i. \end{cases}$$

This a also satisfies the constraint in the quadratic programming (2.5); thus it is the solution of this quadratic programming. Therefore,

$$\tilde{f}(x_i) = \sum_{j=1}^n a_j f(x_j) = f(x_i).$$

□

We also analyze the rate at which the interpolant function $\tilde{f}$ converges to the target function f. We prove that under certain assumptions on the weights w_k in Eq. (2.4) and certain smoothness properties of f, the interpolant converges to the target function at a rate faster than any polynomial order. We first prove a bound for the pointwise approximation error $|\tilde{f}(x) - f(x)|$ when the measurement errors σ_i are 0. Based on this pointwise bound, we then derive a bound for the infinite functional norm of the approximation error $\|\tilde{f} - f\|_\infty$.

Let $\mathcal{Q}^*(x)$ be the unique minimum of the constraint quadratic programming (2.5), and $a_1, \ldots, a_n$ be its solution. From Eq. (2.4) we have

$$\mathcal{Q}^*(x) \geq w_k^2 \left(\sum_{i=1}^{n} a_i \frac{(x_i - x)^k}{k!} \right)^2 , \quad k = 1, \ldots, N \ ,$$

$$\mathcal{Q}^*(x) \geq w_{N+1}^2 \left(a_i \frac{(x_i - x)^{N+1}}{(N+1)!} \right)^2 , \quad i = 1, \ldots, n \ .$$

Incorporating these into the formula for the approximation error (2.3), we get

$$\frac{\left|\tilde{f}(x) - f(x)\right|}{\sqrt{Q^*(x)}} \leq \sum_{k=1}^{N} \frac{|f^{(k)}(x)|}{w_k} + \sum_{i=1}^{n} \frac{|f^{(N+1)}(\xi_i)|}{w_{N+1}} \tag{5.1}$$

As this equation demonstrates, in order to bound the pointwise approximation error, it is critical to bound $Q^*(x)$. The following Lemma provides such a bound for $Q^*(x)$.

LEMMA 3. *Let* $\sigma_i = 0, \forall i$. *Let* $w_k \leq \beta \gamma^k$ *for some* β *and* γ. *Let* $1 \leq p \leq \min(n, N)$. *Let* $\{\hat{x}_1, \ldots, \hat{x}_p\} \subset \{x_1, \ldots, x_n\}$ *satisfies*

$$\max_{1 \leq i \leq p} |x - \hat{x}_i| = D_p \quad \textit{and} \quad \min_{i_1 \neq i_2} |\hat{x}_{i_1} - \hat{x}_{i_2}| \geq \frac{D_p}{r_p} \ .$$

Then the following inequality holds when $\gamma D_p \leq 1$*:*

$$\mathcal{Q}^*(x) \leq \beta^2 \left(\frac{e}{p!^2} + \frac{r_p^{2p}}{N+1} \right) (\gamma D_p)^{2p} \ . \tag{5.2}$$

Proof. Let

$$\hat{a}_i = \frac{\prod_{j \neq i}(x - x_j)}{\prod_{j \neq i}(x_i - x_j)} \ , \quad i = 1, \ldots, p \ ,$$

then for any function g, $\sum_{i=1}^{p} \hat{a}_i g(\hat{x}_i)$ is the value of the Lagrange interpolant of g at x with nodes $\hat{x}_1, \ldots, \hat{x}_p$. The residual of this Lagrange interpolation can be bounded by Boyd (1999)

$$\left| \sum_{i=1}^{p} \hat{a}_i g(\hat{x}_i) - g(x) \right| \leq \frac{\prod_{i=1}^{p} |x - \hat{x}_i|}{p!} \left| g^{(p)}(\xi) \right| \tag{5.3}$$

where $\min(x, \hat{x}_1, \ldots, \hat{x}_p) \leq \xi \leq \max(x, \hat{x}_1, \ldots, \hat{x}_p)$. In particular, let $g(\xi) = (\xi - x)^k$, then $g(x) = 0$. Incorporating this into Eq. (5.3), we have

$$\left|\sum_{i=1}^{p} \hat{a}_i \frac{(\hat{x}_i - x)^k}{k!}\right| = 0 \quad \text{if} \quad k < p,$$

and

$$\left|\sum_{i=1}^{p} \hat{a}_i \frac{(\hat{x}_i - x)^k}{k!}\right| \le \frac{\prod_{i=1}^{p} |x - \hat{x}_i|}{p!} \frac{|\xi - x|^{k-p}}{(k-p)!} \le \frac{1}{p!(k-p)!} D_p^k \quad \text{if} \quad k \ge p.$$

Now let

$$a_i = \begin{cases} \hat{a}_j, & x_i = \hat{x}_j, x_{i'} \ne \hat{x}_j \ \forall\, i' < i \\ 0, & \text{otherwise.} \end{cases}$$

Then

$$\left|\sum_{i=1}^{n} a_i \frac{(x_i - x)^k}{k!}\right| = \left|\sum_{i=1}^{p} \hat{a}_i \frac{(\hat{x}_i - x)^k}{k!}\right| \le \begin{cases} 0\,, & k < p\,, \\ \dfrac{1}{p!(k-p)!} D_p^k\,, & k \ge p\,. \end{cases} \tag{5.4}$$

In addition, from the definition of $\hat{a}_i$, we have

$$|\hat{a}_i| \le \frac{\max\limits_{1 \le i \le p} |x - \hat{x}_i|^p}{\min\limits_{1 \le i \le p} |\hat{x}_{i_2} - \hat{x}_{i_2}|^p} \le r_p^p\,. \tag{5.5}$$

Incorporating Eqs. (5.4) and (5.5) into the definition of $\mathcal{Q}$ in Eq. (2.4) yields

$$\mathcal{Q}(x, a_1, \ldots, a_n) \le \sum_{k=p}^{N} \left(\frac{w_k D_p^k}{p!(k-p)!}\right)^2 + \sum_{i=1}^{p} \left(\frac{w_{N+1} D_p^{N+1}}{(N+1)}\, r_p^p\right)^2\,.$$

Using the assumption $w_k \le \beta\,\gamma^k$ and $p \le N$ leads to

$$\mathcal{Q}(x, a_1, \ldots, a_n) \le \frac{\beta^2 (\gamma D_p)^{2p}}{p!^2}\, e^{(\gamma D_p)^2} + \frac{\beta^2 r_p^{2p}}{N+1}\, (\gamma D_p)^{2N+2}\,.$$

Since $\mathcal{Q}^*(x) \le \mathcal{Q}(x, a_1, \ldots, a_n)$, we have

$$\mathcal{Q}^*(x) \le \frac{\beta^2 (\gamma D_p)^{2p}}{p!^2}\, e^{(\gamma D_p)^2} + \frac{\beta^2 r_p^{2p}}{N+1}\, (\gamma D_p)^{2N+2}\,.$$

When $\gamma D_p \le 1$, $e^{(\gamma D_p)^2} \le e$ and $(\gamma D_p)^{2N+2} \le (\gamma D_p)^{2p}$, the result is

$$\mathcal{Q}^*(x) \le \beta^2 \left(\frac{e}{p!^2} + \frac{r_p^{2p}}{N+1}\right) (\gamma D_p)^{2p}\,.$$

□

This bound of $\mathcal{Q}^*(x)$ naturally leads to a bound of the pointwise approximation error under certain assumptions of the high-order derivative of the function f.

THEOREM 4. *Let $\sigma_i = 0, \forall i$. Let $w_k \le \beta\,\gamma^k$ for some β and γ. Let $1 \le p \le \min(n, N)$. Let $\{\hat{x}_1, \ldots, \hat{x}_p\} \subset \{x_1, \ldots, x_n\}$ satisfies*

$$\max_{1 \le i \le p} |x - \hat{x}_i| = D_p \quad \textit{and} \quad \min_{i_1 \ne i_2} |\hat{x}_{i_1} - \hat{x}_{i_2}| \ge \frac{D_p}{r_p}\,.$$

Let the high-order derivatives of the target function f satisfy

$$\|f^{(k)}\|_\infty \le w_k, \quad k = 1, \ldots, N+1.$$

Then the following inequality holds when $\gamma D_p \le 1$:

$$|\tilde{f}(x) - f(x)| \le (N+n)\,\beta\,\sqrt{\frac{e}{p!^2} + \frac{r_p^{2p}}{N+1}}\,(\gamma D_p)^p\ .$$

Proof. This result is obtained by incorporating Eq. (5.2) into Eq. (5.1). □

Note that in order to obtain this pointwise error bound, the high-order derivatives of f must be bounded by w_k, which is in turn bounded by $\beta\gamma^k$. This limits this result to functions that are smooth enough, so that the high-order derivatives grow at most exponentially fast. With this pointwise error bound, we can construct a global error bound.

LEMMA 4. *Let $\sigma_i = 0, \forall i$. Let $w_k \le \beta\,\gamma^k$ for some β and γ. Let $n \ge N \ge p$; the nodes*

$$b_l = x_1 < x_2 < \ldots < x_n = b_u$$

are well-spaced, so that

$$\frac{\max\limits_{1\le i<n} x_{i+1} - x_i}{\min\limits_{1\le i<n} x_{i+1} - x_i} \le r.$$

Let the high-order derivatives of the target function f satisfy

$$\|f^{(k)}\|_\infty \le w_k, \quad k = 1, \ldots, N+1.$$

Then the following inequality holds when $n \ge \gamma\,p\,r\,(b_u - b_l) - p\,(r-1)$:

$$\|\tilde{f} - f\|_\infty = \max_{b_l \le x \le b_u} |\tilde{f}(x) - f(x)| \le 2\beta\,\sqrt{\frac{e}{p!^2} + n\frac{(r\,p)^{2p}}{N+1}}\;n\left(\gamma\frac{p\,r\,(b_u - b_l)}{n + p\,(r-1)}\right)^p\ . \quad (5.6)$$

Proof. Since

$$\frac{\max\limits_{1\le i<n} x_{i+1} - x_i}{\min\limits_{1\le i<n} x_{i+1} - x_i} \le r,$$

for any $b_l \le x \le b_u$, let $\hat{x}_i, i = 1, \ldots, p$ be the p grids closest to x, then

$$D_p \max_{1\le i\le p} |x - \hat{x}_i| \le \frac{p\,r\,(b_u - b_l)}{n + p\,(r-1)}\ .$$

Also,

$$n \ge \gamma\,p\,r\,(b_u - b_l) - p\,(r-1) \Longrightarrow \gamma D_p \le 1.$$

Conversely, since

$$\frac{\max\limits_{1\le i<n} x_{i+1} - x_i}{\min\limits_{1\le i<n} x_{i+1} - x_i} \le r,$$

we have

$$\min_{i_1 \ne i_2} |\hat{x}_{i_1} - \hat{x}_{i_2}| \ge \frac{D_p}{r_p}\ ,$$

where

$$r_p = r\,p.$$

Using the fact that $N \leq n$, Theorem 4 yields

$$|\tilde{f}(x) - f(x)| \leq 2\beta \sqrt{\frac{e}{p!^2} + \frac{(r\,p)^{2p}}{N+1}}\; n \; \left(\gamma\, \frac{p\,r\,(b_u - b_l)}{n + p\,(r-1)}\right)^p$$

for any $b_l \leq x \leq b_u$. □

When n is large, the r.h.s. of Eq. (5.6) is $O(n^{p-\frac{3}{2}})$. Therefore, this lemma shows that for a fixed N, the interpolation approximation $\tilde{f}$ converges to the target function f at a polynomial rate of $p - \frac{3}{2}$ for any $p \leq N$. This result is similar to the rate of convergence of Floater-Hormann rational interpolation Floater & Hormann (2007). In addition, the following theorem demonstrates a further consequence of this lemma.

Theorem 5. *In addition to the assumptions in Lemma 4, if $N = n$, then*

$$\|\tilde{f} - f\|_\infty \leq M\, n \left(\gamma \frac{b-a}{n + p(r-1)}\right)^p . \tag{5.7}$$

where

$$M = 2\beta \sqrt{\frac{e}{p!^2} + (pr)^{2p}}$$

is independent of n.

Proof. It follows directly from Lemma 4. □

Because the r.h.s. of Eq. (5.7) is $O(n^{p-1})$ when n is large, this corollary reveals the rate at which our interpolant approximation $\tilde{f}$ converges to the target function f as the number of nodes n increases. Assume that f is smooth enough so that its high-order derivatives satisfy $\|f^{(k)}\|_\infty \leq w_k$ for all $k > 0$, the grid is well spaced so that the ratio between the largest interval and smallest interval is bounded by r, and we use $N = n$ in our scheme. Then for any positive integer p, when n is big enough, i.e., $n \geq \max(p, \gamma\, p\, r\,(b-a) - p\,(r-1))$, the approximation error converges at a polynomial rate of order $p - 1$. This implies that under these assumptions, the rate of convergence of our interpolation approximation is faster than any polynomial order. This theoretical result is verified by numerical experiments. We show in Sec. 7 that the rate of convergence is exponentially fast for all smooth functions attempted.

6. Choice of the weights

From Eq. (5.1), it is clear that the approximation error of $\tilde{f}$ is a weighted sum of the derivatives of f of various orders. w_k controls the weight of the contribution from the kth derivative. This section discusses one possible choice of the weights w_k in our interpolation approximation scheme. We choose this method of calculating the weights because it yields good results experimentally. Therefore, some formulas in this section are obtained empirically without rigorous mathematical analysis.

Theorem 4 and Theorem 5 places the following restrictions on the weights w_k:

(a) The high-order derivatives of f are bounded by w_k.

(b) w_k is bounded by $\beta\gamma^k$ for some β and γ.

For this reason, we choose the weights w_k to be

$$w_k = \beta \gamma^k, \tag{6.1}$$

and compute the two parameters β and γ using the datapoints. The rest of this section discusses how these two parameters are evaluated, and how these parameters affect the approximation function $\tilde{f}$.

We first consider β. We call this parameter the "magnitude" of the target function f. This is because it is supposed to bound the magnitude of the target function in Theorem 4 and 5.

$$\beta = w_0 \geq \|f^{(0)}\|_\infty .$$

In reality, we find that our interpolation approximation is more accurate if we choose a smaller β. Based on this fact, we can estimate β by taking the sample standard deviation of all the given datapoints, i.e.,

$$\beta = \sqrt{\frac{\sum_{i=1}^n (f(x_i) - \bar{f}\,)^2}{n-1}} \;, \qquad \text{where } \bar{f} = \frac{\sum_{i=1}^n f(x_i)}{n} \;.$$

In the presence of measurement errors, an estimate of β is.

$$\beta = \sqrt{\frac{\sum_{i=1}^n (\hat{f}(x_i) - \bar{f}\,)^2}{n-1} \;\exp\left(-\frac{n-1}{n} \frac{\sum_{i=1}^n \sigma_i^2}{\sum_{i=1}^n (\hat{f}(x_i) - \bar{f}\,)^2} \right)} \;. \tag{6.2}$$

The "magnitude" parameters β augments the relative importance of the given measurements to the importance of the measurement errors, and determines how hard the interpolant tries to pass through each data point. By observing the quadratic form Eq. (2.4) with $w_k = \beta\gamma^k$, we find that β has no effect on the interpolant when there are no measurement errors, i.e., $\sigma_i = 0, i = 1, \ldots, n$. When measurement errors are present, the ratio of $e_{a\,i}$ to β presents a relation, between the contribution to the variation of $\hat{f}(x_i)$ from the measurement errors, and the contribution to the variation of $\hat{f}(x_i)$ from the variation of the function f itself. This can be seen from the composition of the quadratic form (2.4), as β determines the relative importance of the first two terms in A relative to the third term, which corresponds to the measurement errors in $\hat{f}$. When β is small compared to the measurement errors, $A \approx H^2$, and the interpolant $\tilde{f}$ is a constant function whose value is

$$\tilde{f}(z) \equiv \sum_{i=1}^n \frac{1}{\sigma_i^2} \hat{f}(x_i).$$

In this case, all the variation of the function values $\hat{f}(x_i)$ are attributed to the measurement errors, and $\tilde{f}$ makes no effort to go through each datapoint. On the other hand, when β is large compared to the measurement errors, the third term in the definition of A can be ignored compared to the first two terms. As discussed in the previous section, the interpolant $\tilde{f}$ fits the datapoint $\hat{f}(x_i)$ exactly when $\sigma_i = 0$. In this case, all the variation of the function values $\hat{f}(x_i)$ are attributed to the variation of the function f itself.

The other parameter, the "roughness" γ, models how fast the kth derivative of f grows as k increases. γ is called the "roughness" because if f is a sine wave of angular frequency γ, i.e., $f(x) = e^{i\gamma x}$, then γ is the rate of growth of its derivative, i.e., $\|f^{(k)}(x)\| = \gamma^k \|f(x)\|$. We use this as a model for general smooth functions by assuming that the bound of the kth derivative of f grows exponentially as γ^k. Here the parameter γ describes the frequency of the fastest varying mode of the function f, or the reciprocal of the smallest length scale of f. With an appropriate choice of γ, this model provides a valid estimate of the magnitude of $f^{(k)}(z)$ for most smooth functions.

We determine γ from the given data using the bi-section method. Upper and lower bounds are first determined from the spacing of the nodes, then the interval of possible γ is bisected by interpolating each datapoint using other datapoints with our method, and comparing the actual residual $f(x_i) - \tilde{f}(x_i)$ with $\mathcal{Q}^*(x_i)$. In determining the upper and lower bound, we rely on the fact that the reciprocal of γ models the smallest length scale of f. On the other hand, the possible length scales that can be reconstructed from the finite number of datapoints are limited by the span of the datapoints on one end, and by the Nyquist sampling theorem on the other end. Specifically, we start the bi-section with

$$\gamma_{\text{min}} = \frac{1}{\delta_{\text{max}}} , \quad \gamma_{\text{max}} = \frac{\pi}{\delta_{\text{min}}} ,$$

where δ_{max} and δ_{min} are the maximum and minimum distance between any two nodes. The interval $[\gamma_{\text{min}}, \gamma_{\text{max}}]$ is then bisected logarithmically at each step by $\gamma_{\text{mid}} = \sqrt{\gamma_{\text{min}}\gamma_{\text{max}}}$. With this γ_{mid}, for each $i = 1, \ldots, n$, we use our interpolation scheme to calculate $\tilde{f}(x_i)$ with datapoints other than the one at x_i. We then compare $\mathcal{Q}^*(x_i)$ calculated with Eq. (2.5) with the true residual $r(x_i) = \tilde{f}(x_i) - f(x_i)$. The bi-section algorithm decide that $\gamma < \gamma_{\text{mid}}$ if $\mathcal{Q}^*(x_i), i = 1, \ldots, n$ are larger than the true residuals, or $\gamma > \gamma_{\text{mid}}$ otherwise. This choice is based on the observation that a larger γ results in a larger $\mathcal{Q}^*(x_i)$. Specifically, we set

$$\begin{aligned} \gamma_{\text{max}} &= \gamma_{\text{mid}} \quad \text{if} \quad \frac{1}{n}\sum_{i=1}^{n} \frac{r(x_i)^2}{\mathcal{Q}^*(x_i)} < 1 , \\ \gamma_{\text{min}} &= \gamma_{\text{mid}} \quad \text{if} \quad \frac{1}{n}\sum_{i=1}^{n} \frac{r(x_i)^2}{\mathcal{Q}^*(x_i)} > 1 , \end{aligned}$$

or when σ_i are non-zero,

$$\begin{aligned} \gamma_{\text{max}} &= \gamma_{\text{mid}} \quad \text{if} \quad \frac{1}{n}\sum_{i=1}^{n} \frac{\left(\tilde{f}(x_i) - \hat{f}(x_i)\right)^2}{\mathcal{Q}^*(x_i) + \sigma_i^2} < 1 , \\ \gamma_{\text{min}} &= \gamma_{\text{mid}} \quad \text{if} \quad \frac{1}{n}\sum_{i=1}^{n} \frac{\left(\tilde{f}(x_i) - \hat{f}(x_i)\right)^2}{\mathcal{Q}^*(x_i) + \sigma_i^2} > 1 . \end{aligned} \tag{6.3}$$

The bi-section continues until γ_{min} and γ_{max} are sufficiently close. We stop the bi-section when

$$\frac{\gamma_{\text{max}}}{\gamma_{\text{min}}} < T_\gamma$$

for some threshold T_γ. At this point, we use γ_{mid} as the estimation for the "roughness" parameter γ. Through numerical experiments several different functions, we found that $T_\gamma \approx 1.1$ is enough to produce very good results.

The parameter γ determines how aggressively the interpolant tries to achieve polynomial accuracy. As can be seen from Eq. (5.1) with $w_k = \beta\gamma^k$, when γ is small, high-order derivatives contribute more to the approximation error. In this case, the interpolation is accurate on smooth functions such as lower-order polynomials, but may produce very large errors if the function contains small length scales, which cause its high-order deriva-

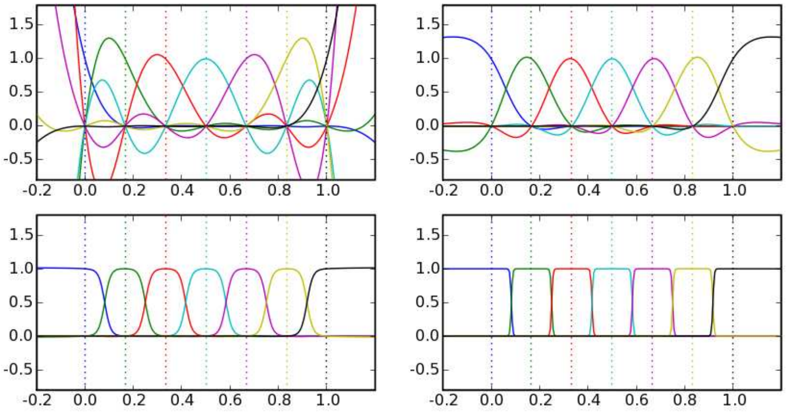

FIGURE 1. The basis functions on uniform grid for $\gamma = 1$ (upperleft), $\gamma = 10$ (upperright), $\gamma = 25$ (lowerleft) and $\gamma = 100$ (lowerright). The dotted vertical lines indicates the location of the uniformly spaced nodes, and each solid line of corresponding color is the unique interpolant that equals 1 at that node and 0 at all other nodes.

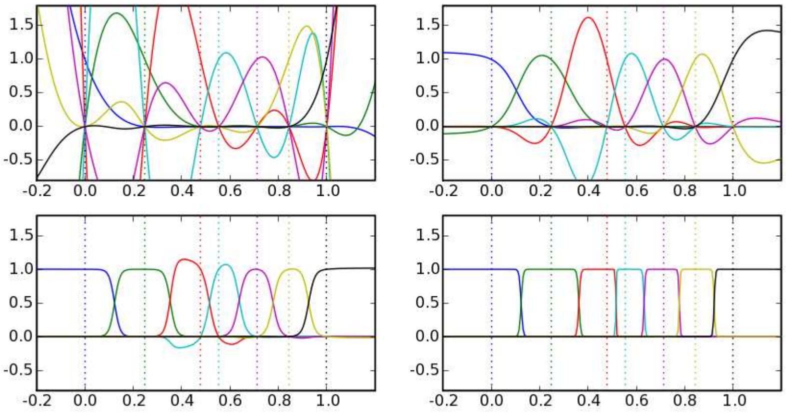

FIGURE 2. The basis functions on non-uniform grid for different roughness γ: $\gamma = 1$ (upperleft), $\gamma = 10$ (upperright), $\gamma = 25$ (lowerleft) and $\gamma = 100$ (lowerright). The dotted vertical lines indicates the location of the non-uniformly spaced nodes, and the each solid line of corresponding color is the unique interpolant that equals 1 at that node and 0 at all other nodes.

tives to grow rapidly. As γ increases, the approximation error results more and more from low-order derivatives. The interpolation becomes more robust on oscillatory functions but less accurate on smooth functions.

To illustrate the effects of γ, Figures 1 and 2 plots the basis of the interpolant for different values of γ on uniform and non-uniform grids. Both the uniform and non-uniform grids consist of 7 nodes. The basis of the interpolant at each node is defined as the unique interpolant on this grid that equals 1 on that node and 0 on all other nodes. An interpolant constructed from any function values given on this grid is a linear combination of these basis functions. Two important changes are observed as γ varies. First, the support of the basis functions increases as γ decreases. Although our interpolation scheme is formally global, i.e., the function value at each data point influences the interpolant globally, the area where the influence is essentially non-zero is finite when γ is large. In both the uniform and non-uniform cases, the effective support of each basis function when $\gamma = 100$ barely covers the nearest neighborhood of the corresponding node. As γ decreases to 25, the supports of each basis function extends to neighboring nodes, sometimes beyond a neighboring node in the non-uniform grid. When γ further reduces to 10, the support of each basis covers multiple nodes. When $\gamma = 1$, the basis really become global functions without a finite support.

The second change when γ decreases is the increase of the Lebesgue constant. The Lebesgue constant Λ is defined as the operator norm of the interpolation scheme as a linear mapping from the space of continuous functions to itself, i.e.,

$$\Lambda = \sup_{||f||=1} \left\| \tilde{f} \right\| ,$$

where $\| \cdot \|$ is the maximum norm. It can be shown that Λ is equal to the maximum of all basis functions within the interpolation interval. Since the interpolant must go through each datapoint, the Lebesgue constant is greater or equal to 1. As shown in Figures 1 and 2, the basis functions are almost capped at 1 when $\gamma = 100$, and the Lebesgue constant is very close to unity. As γ decreases, the basis functions overshoots higher above 1, and the Lebesgue constant increases. When $\gamma = 1$, the Lebesgue constant is approximately 1.3 in the uniform grid, and above 2 in the non-uniform grid. We also notice that for the same γ, the Lebesgue constant is higher for the non-uniform grid.

These two effects, the increase of the Lebesgue number, and the growth of the support of each basis function as γ decreases, dominates the behavior of the interpolant. A smaller γ generates a more global set of basis functions, allowing the use of a larger number of datapoints in the calculation of the interpolant value, resulting in a more accurate approximation for smooth functions. A larger γ, on the other hand, represents a more conservative approach. By using fewer datapoints to determine the value of the interpolant, the interpolation scheme loses high-order accuracy for smooth functions; however, by constraining the basis functions to a more local support, it has a smaller Lebesgue constant, making it more robust for non-smooth functions.

These effects of γ can be further demonstrated by the extreme cases when γ is very small and very large. In Appendix A, we prove that our interpolation scheme converges to Lagrange polynomial interpolation in the limit of $\gamma \to 0^+$. In this case, it converges exponentially fast for smooth functions on a good grid, but is not robust for oscillatory functions on a bad grid. On the other hand, when $\gamma \to +\infty$, it converges to the Shepard's interpolation Gordon & Wixom (1978), which is robust, but has slow convergence.

7. Numerical solution of the interpolation coefficients

In this section we discuss how to solve the quadratic programming problem (2.5). Due to the large condition number of the matrix A, directly solving the linear system (3.5) produces large numerical errors that completely corrupt the solution. Therefore, our goal is to calculate the coefficients a without explicitly constructing A.

We define

$$\hat{a} = -\frac{2a}{\lambda} ,$$

then Eq. (3.5) becomes

$$A\,\hat{a}^{\mathrm{T}} = e^{\mathrm{T}}, \quad a = \frac{\hat{a}}{e\,\hat{a}^{\mathrm{T}}}\;.$$

Therefore, it is sufficient to accurately solve $A\,\hat{a}^{\mathrm{T}} = e^{\mathrm{T}}$. Let V be an N by n matrix with

$$V_{ki} = w_k \frac{(x_i - x)^k}{k!},$$

F be an n by n diagonal matrix with

$$E_{ii}^2 = \left(\frac{(x_i - x)^{N+1}}{(N+1)!}\right)^2 + \sigma_i^2\;,$$

then

$$A = V^{\mathrm{T}}V + E^2 = \left[V^{\mathrm{T}}E^{\mathrm{T}}\right]\begin{bmatrix} V \\ E \end{bmatrix}.$$

We perform a QR decomposition of the matrix

$$\begin{bmatrix} V \\ E \end{bmatrix} = QR, \tag{7.1}$$

so that Q is an orthonormal matrix, and R is an upper-triangular matrix. Then the linear system we want to solve becomes

$$R^{\mathrm{T}}R\,\hat{a}^{\mathrm{T}} = e^{\mathrm{T}}.$$

Therefore, the interpolation coefficients can be solved by one forward elimination, one backward elimination and a normalization:

$$R^{\mathrm{T}}\hat{\hat{a}}^{\mathrm{T}} = e^{\mathrm{T}}, \quad R\,\hat{a}^{\mathrm{T}} = \hat{\hat{a}}, \quad a = \frac{\hat{a}}{e\,\hat{a}^{\mathrm{T}}}\;. \tag{7.2}$$

This process proves to be much more stable than directly solving Eq. (3.5) and produces sufficiently accurate results. With this algorithm, he total number of operations involved in constructing the interpolant at each point is Golub & Loan (1996) $O((N+n)n^2)$.

8. Numerical examples

In this section, we apply our interpolation scheme to the following example functions:

(*a*) A cosine wave $f(x) = \cos x$. This function is smooth and expected to be an easy case for interpolation schemes.

(*b*) The Runge function $f(x) = \dfrac{1}{1+x^2}$. It was used by Runge (1901) to demonstrate the divergence of Lagrange interpolation on an equally spaced grid.

(*c*) A cosine wave with a sharp Gaussian notch $f(x) = \cos x - 2e^{-(4x)^2}$. Since this function contains two distinct length scales, we use it as a simple model for multi-scale functions.

(*d*) A discontinuous function $f(x) = \begin{cases} e^{-x^2}, & x > 0, \\ -e^{-x^2}, & x < 0. \end{cases}$.

For all these functions, the interpolant is constructed in the interval $[-5, 5]$ using two kinds of grids: a uniformly distributed grid and a Niederreiter quasi-random sequence grid Niederreiter (1992). Figures 3 to 10 demonstrate the performance of our interpolation scheme on these four functions using both the uniform and quasi-random grid. In each

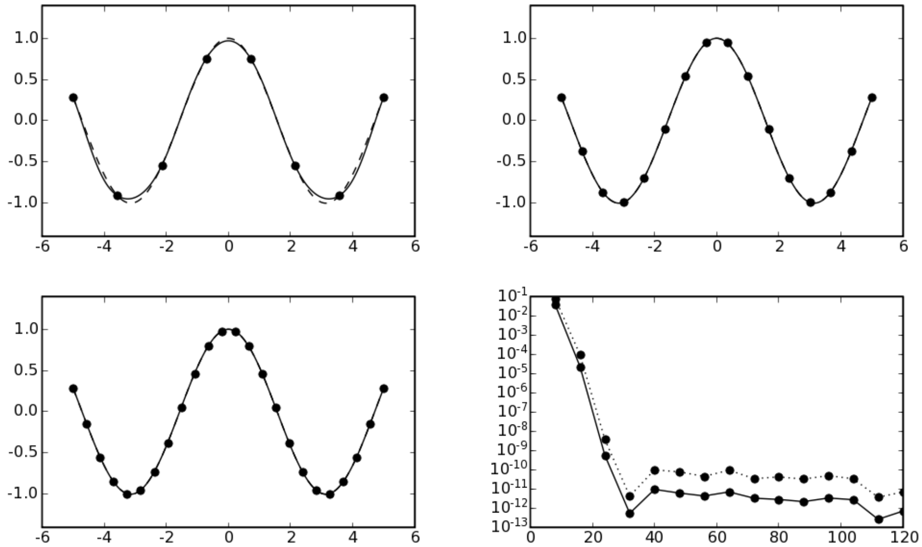

FIGURE 3. Interpolating the cosine wave using 8 (upperleft), 16 (upperright), 24 (lowerleft) uniform grid, and the convergence plot (lowerright).

figure, the first three plots show the target function as dashed lines and our interpolant approximation as solid lines. The dots indicate the location of the datapoints. The fourth plot in each figure shows the convergence of our interpolations scheme. The horizontal axis is the number of datapoints used, and the vertical axis is the difference between the target function and the interpolant function, measured in L^∞ distance (dotted lines) and in L^2 distance (solid lines). As can be seen from these figures, our interpolation scheme works robustly for all four functions on both uniform and quasi-random grids. For the three smooth functions, it converges exponentially to a cutoff precision of approximately 10^{-10} to 10^{-13}. The rate of convergence is fastest for the cosine wave, and slowest for the multi-scale notched cosine function. This behavior is expected because the notched cosine function contains the most high frequency components, while the plain cosine function contains the least. The cut-off precision is due to the round-off error accumulated in the QR factorization (7.1) and solution of the linear systems (**??**). For the discontinuous function, we observe artificial oscillations near the discontinuity, whose size doesn't seem to decrease as the grid refines. As a result, the L^∞ error in the convergence plots stays almost constant, and the L^2 error decreases slowly. Despite of this Gibbs-like phenomenon, the interpolant seems to converge point-wise to the target function, just as Lagrange interpolation does on a Lobatto grid. In these numerical experiments, our interpolation demonstrates high accuracy for smooth functions, and excellent robustness even for discontinuous functions. It combines the accuracy of Lagrange interpolation on Lobatto grid and the flexibility and robustness of Shepard's interpolation.

Figure 11 demonstrates our approximation scheme when measurement errors are nonzero. In this case, our scheme becomes a nonlinear regression scheme by constructing a "best fit" for the data. The convergence plot shows slow convergence to the target function, due to the corruption by the measurement errors.

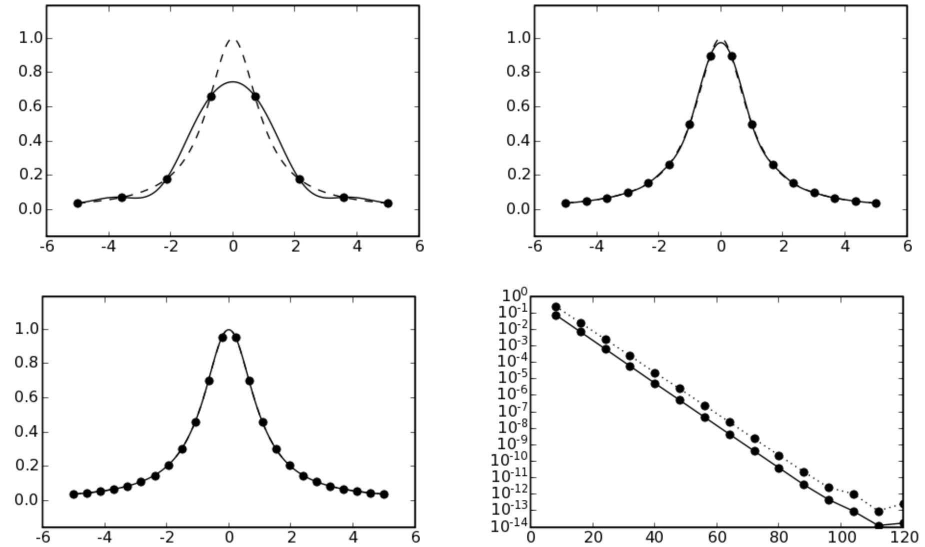

FIGURE 4. Interpolating the Runge function using 8 (upperleft), 16 (upperright), 24 (lowerleft) uniform grid, and the convergence plot (lowerright).

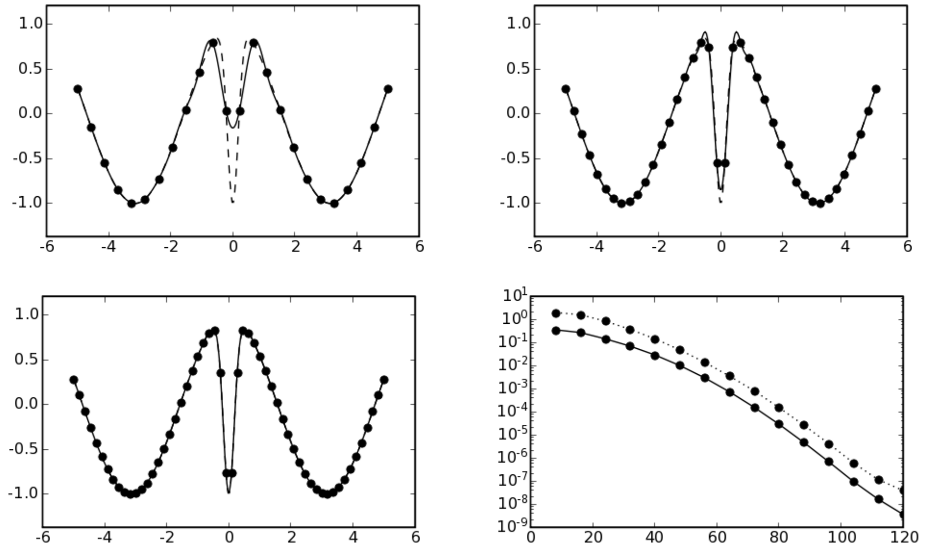

FIGURE 5. Interpolating the notched cosine function using 24 (upperleft), 40 (upperright), 56 (lowerleft) uniform grid, and the convergence plot (lowerright).

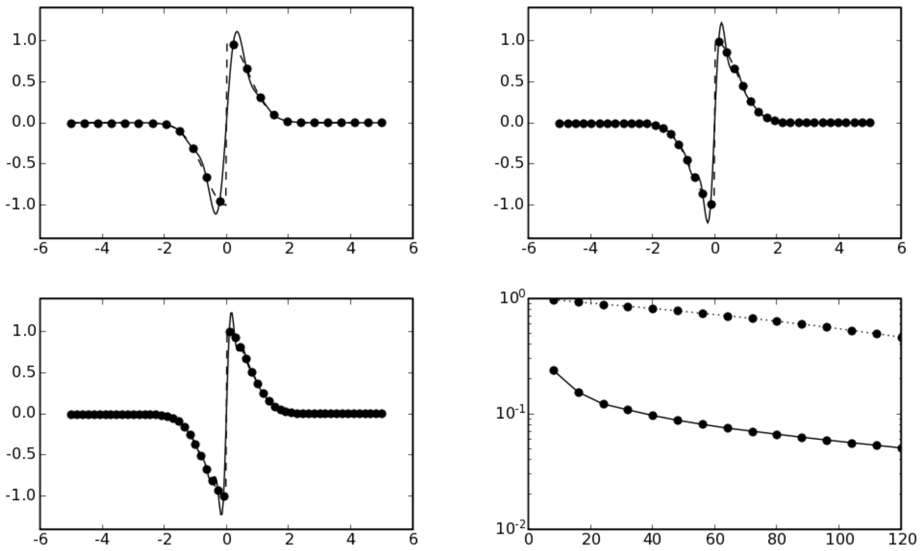

FIGURE 6. Interpolating the discontinuous function using 24 (upperleft), 40 (upperright), 56 (lowerleft) uniform grid, and the convergence plot (lowerright).

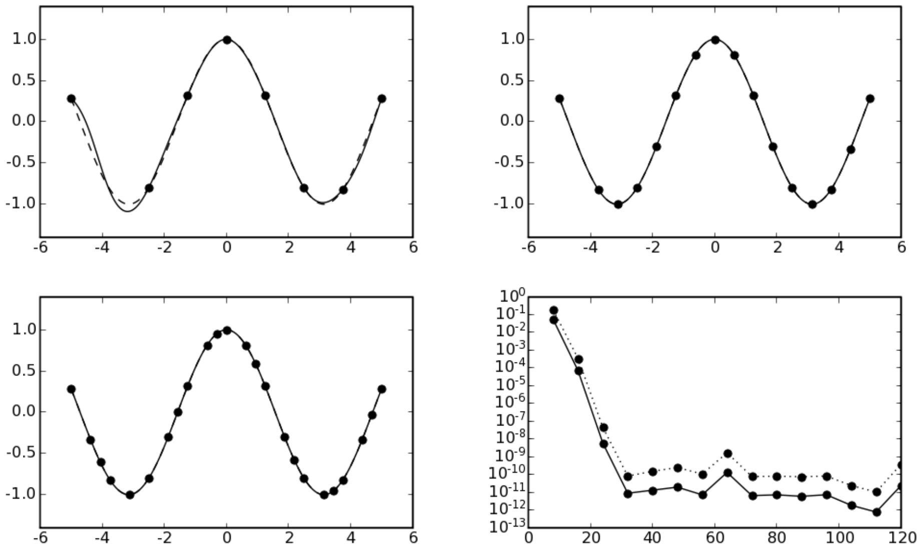

FIGURE 7. Interpolating the cosine wave using 8 (upperleft), 16 (upperright), 24 (lowerleft) quasi-random grid, and the convergence plot (lowerright).

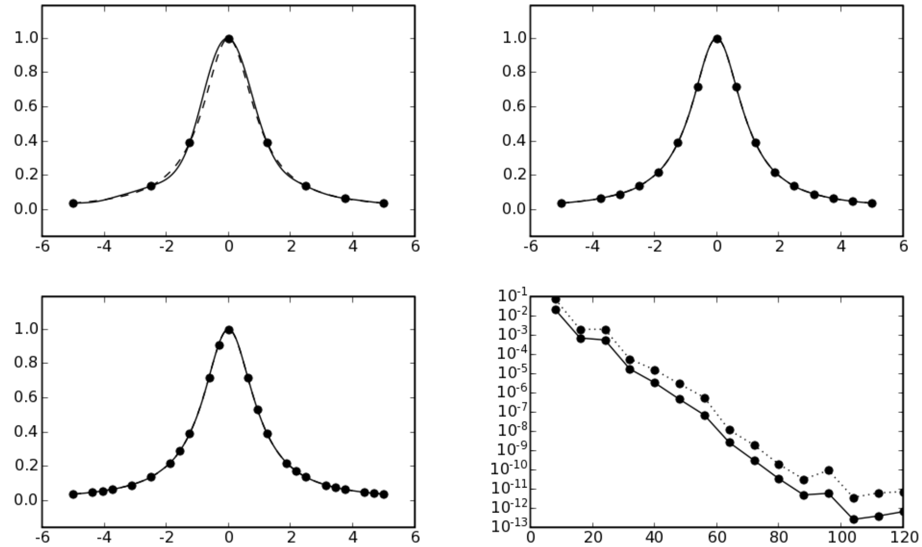

Figure 8. Interpolating the Runge function using 8 (upperleft), 16 (upperright), 24 (lowerleft) quasi-random grid, and the convergence plot (lowerright).

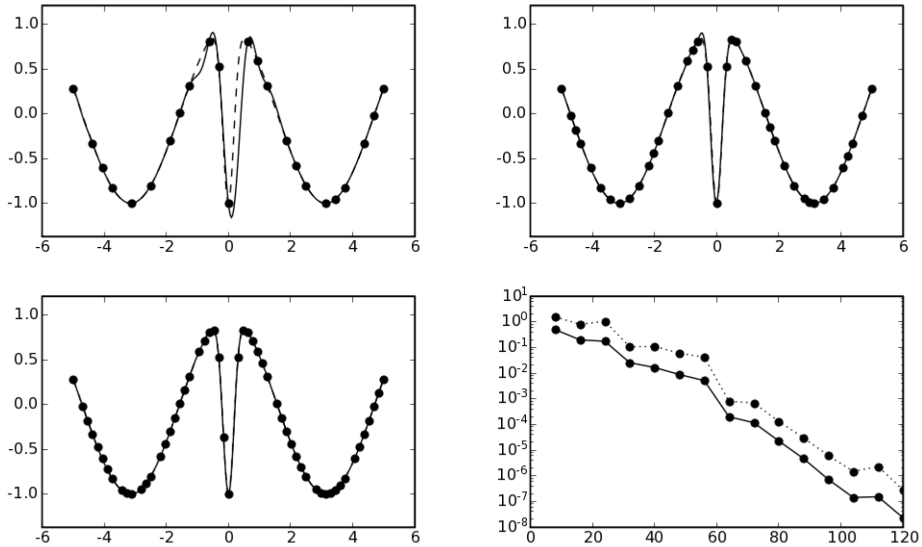

Figure 9. Interpolating the notched cosine function using 24 (upperleft), 40 (upperright), 56 (lowerleft) quasi-random grid, and the convergence plot (lowerright).

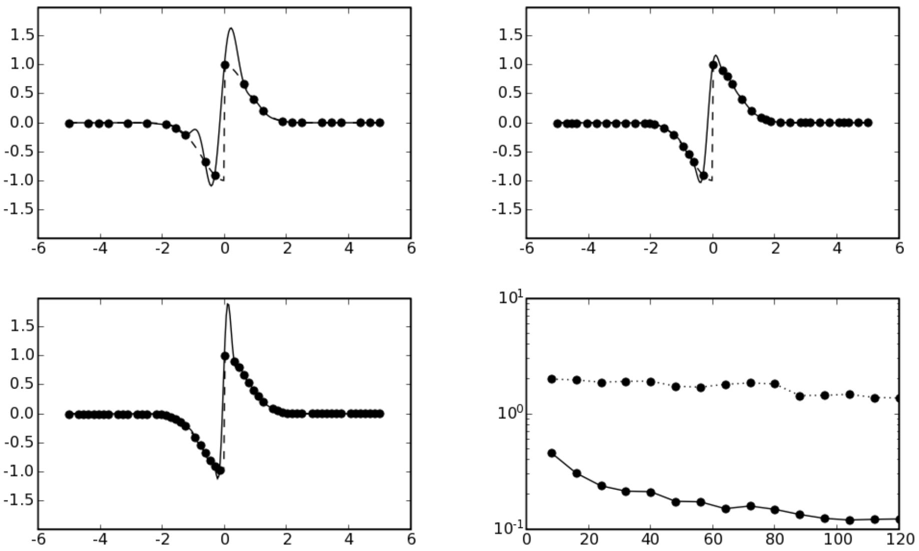

FIGURE 10. Interpolating the discontinuous function using 24 (upperleft), 40 (upperright), 56 (lowerleft) quasi-random grid, and the convergence plot (lowerright).

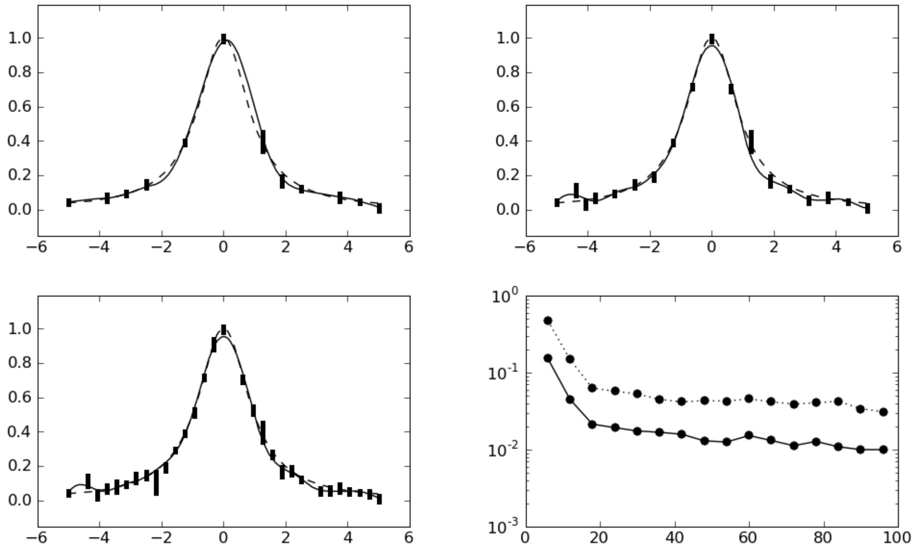

FIGURE 11. Interpolating the Runge function using 12 (upperleft), 18 (upperright), 30 (lowerleft) quasi-random grid with measurement errors indicated by bars. The lowerright is the convergence plot.

9. Conclusion

A univariate interpolation scheme on arbitrary grid is developed. The interpolant is a rational function with no poles on the real line. With proper choice of the weights w_k, the interpolant converges to a smooth target function at a super-polynomial rate. In addition, when errors exist on data points, the interpolation scheme becomes a nonlinear regression scheme. Experiments of this interpolation scheme on uniform and non-uniform grids show that it has both exponential convergence on smooth functions, and excellent robustness on non-smooth and discontinuous functions.

Future research in several areas are currently being conducted. First, it is easy to generalize the mathematical formulation of this interpolation scheme to multivariate case. We are working on proving similar properties of the multivariate interpolant. Second, if the derivatives are given at the data points, it is possible to construct a Hermite type of interpolation using the same formulation, in which the derivatives at the data points are expanded using a Taylor series similar to the values. Third, when measurement errors are present, our interpolation scheme becomes a nonlinear regression scheme. We want to further study the behavior and properties of this regression scheme. We also want to extend this regression scheme to account for correlated errors and errors with given probability density function.

Acknowledgments

This work was funded by the United States Department of Energy's PSAAP at Stanford University.

REFERENCES

Berrut, J. & Trefethen, L. 2004 Barycentric Lagrange interpolation. *SIAM Rev.* **46-3**, 501–517.

de Boor, C. 2001 *A practicle guide to splines, revised ed.*. New York, NY: Springer-Verlag.

Boyd, J. 1999 A numerical comparison of seven grids for polynomial interpolation on the interval. *Computers and Mathematics with Applications* **38**, 35–50.

Floater, M. & Hormann, K. 2007 Barycentric rational interpolation with no poles and high rates of approximation. *Numerische Mathematik* **107**, 315–331(17).

Golub, G. & Loan, C. V. 1996 *Matrix computations (3rd ed.)*. Baltimore, MD: Johns Hopkins University Press.

Gordon, W. & Wixom, J. 1978 Shepard's method of "metric interpolation" to bivariate and multivariate interpolation. *Math. Comp.* **32-141**, 253–264.

Lsaacson, E. & Keller, H. 1996 *Analysis of Numerical methods*. New York, NY: Dover.

Moin, P. 2001 *Fundamentals of engineering numerical analysis*. New York, NY: Cambridge University Press.

Niederreiter, H. 1992 *Random number generation and quasi-Monte Carlo methods*. Philadelphia, PA: Society for Industrial and Applied Mathematics.

Nocedal, J. & Wright, S. 2000 *Numerical optimization*. New York, NY: Springer.

Runge, C. 1901 Uber empirische funktionen und die interpolation zwischen aquidistanten ordinaten. *Zeitschrift fur Mathematik und Physik* **46**, 224–243.

Appendix A. Miscellaneous facts about the interpolant

Theorem 6. *For a fixed set of data points, $\sigma_i = 0, \forall i$. Let $w_k = \beta\gamma^k$, and the interpolant function $\tilde{f}$ is constructed by Eqs. (2.1) and (2.5) with $N \geq n-1$. Then*

$$\lim_{\gamma\to 0^+} \tilde{f}(x) = f^{\mathcal{L}}(x), \quad \forall x \ ,$$

where $f^{\mathcal{L}}$ is the Lagrange polynomial interpolant on the same data points.

Proof. The Lagrange interpolation

$$f_{\mathcal{L}}(x) = \sum_{i=1}^{n} a_i^{\mathcal{L}} f(x_i), \qquad \text{where} \quad a_i^{\mathcal{L}} = \frac{\prod_{j\neq i}(x-x_j)}{\prod_{j\neq i}(x_i-x_j)}.$$

It is sufficient to prove that

$$\lim_{\gamma\to 0^+} a_i = a_i^{\mathcal{L}}, \quad i = 1,\ldots,n, \tag{A 1}$$

for any x, where a_i are the solution of Eq. (2.5).

Lagrange interpolation $\sum_{i=1}^n a_i f(x_i)$ is exact for polynomials up to order $n-1$. Therefore,

$$\sum_{i=1}^{n} a_i^{\mathcal{L}} (x_i - x)^k = (x-x)^k = \begin{cases} 1, & k=0 \\ 0, & k=1,\ldots,n-1 \ , \end{cases} \tag{A 2}$$

Incorporating Eq. (2.4) and the assumptions $N \geq n-1$, $\sigma_i = 0$ and $w_k = \beta\gamma^k$, we get

$$\begin{aligned} \mathcal{Q}(x, a_1^{\mathcal{L}}, \ldots, a_n^{\mathcal{L}}) = \beta^2 \sum_{k=n}^{N} \gamma^{2k} \left(\sum_{i=1}^{n} a_i^{\mathcal{L}} \frac{(x_i-x)^k}{k!} \right)^2 \\ + \beta^2 \sum_{i=1}^{n} \gamma^{2N+2} \left(a_i^{\mathcal{L}} \frac{(x_i-x)^{N+1}}{(N+1)!} \right)^2 \ . \end{aligned}$$

Therefore,

$$\lim_{\gamma\to 0^+} \frac{\mathcal{Q}(x, a_1^{\mathcal{L}}, \ldots, a_n^{\mathcal{L}})}{\gamma^{2n-1}} = 0 \ .$$

Since $a_i^{\mathcal{L}}$ satisfy the constraint $\sum_{i=1}^n a_i^{\mathcal{L}} = 1$, we know that

$$\mathcal{Q}^*(x) = \mathcal{Q}(x, a_1, \ldots, a_n) \leq \mathcal{Q}(x, a_1^{\mathcal{L}}, \ldots, a_n^{\mathcal{L}}) \ ,$$

therefore,

$$\lim_{\gamma\to 0^+} \frac{\mathcal{Q}(x, a_1, \ldots, a_n)}{\gamma^{2n-1}} = 0 \ .$$

i.e. $\forall\epsilon, \exists\gamma_\epsilon$ s.t. $\forall\, 0 < \gamma < \gamma_\epsilon$,

$$\frac{\mathcal{Q}(x, a_1, \ldots, a_n)}{\gamma^{2n-1}} < \epsilon^2 \ .$$

On the other hand, with out loss of generality, we can assume $\gamma_\epsilon \leq 1$ and $n!^2\gamma_\epsilon \leq \beta^2$. From Eq. (2.4) we obtain

$$\epsilon^2 > \frac{\mathcal{Q}(x, a_1, \ldots, a_n)}{\gamma^{2n-1}} \geq \sum_{k=1}^{n-1} \left(\sum_{i=1}^{n} a_i (x_i - x)^k \right)^2 \ .$$

Denote the n by n Vandermonde matrix $V_{ki} = (x_i - x)^{k-1}$, Then the inequality above combined with the constraint (2.2) implies

$$\left\| V\, a^{\mathrm{T}} - e_1 \right\|_2 < \epsilon \ ,$$

where $a = (a_1, \ldots, a_n)$, $e_1 = (1, 0, \ldots, 0)$. On the other hand, the matrix form of Eq. (A 2) is

$$V\, a^{\mathcal{L}\,\mathrm{T}} = e_1 \ ,$$

where $a^{\mathcal{L}} = (a_1^{\mathcal{L}}, \ldots, a_n^{\mathcal{L}})$. By combining the above inequality and equality, we obtain

$$\left\| V (a^{\mathrm{T}} - a^{\mathcal{L}\mathrm{T}}) \right\|_2 < \epsilon \ .$$

Since the determinant of the Vandermonde matrix V is nonzero Golub & Loan (1996), it is invertible and the operator norm $\|V^{-1}\|_2$ is finite. Therefore, $\forall \epsilon, \exists \gamma_\epsilon$, s.t. $\forall\, 0 < \gamma < \gamma_\epsilon$

$$|a_i - a_i^{\mathcal{L}}| \le \|a - a^{\mathcal{L}}\|_2 \le \|V^{-1}\|_2 \left\| V (a^{\mathrm{T}} - a^{\mathcal{L}\mathrm{T}}) \right\|_2 \le \epsilon\, \|V^{-1}\|_2 \ ,$$

Therefore, Eq. (A 1) holds. □

Theorem 7. *For a fixed set of data points, $\sigma_i = 0, \forall i$. Let $w_k = \beta\gamma^k$, and the interpolant function $\tilde{f}$ is constructed with (2.1) and (2.5). Then*

$$\lim_{\gamma \to +\infty} \tilde{f}(x) = f^{\mathcal{S}}(x), \quad \forall x \ ,$$

where $f^{\mathcal{S}}$ is the Shepard's interpolant Gordon & Wixom (1978) with power parameter $2N + 2$ on the same data points.

Proof. Shepard's interpolation, a.k.a. inverse distance weighting, is given by

$$f_{\mathcal{S}}(x) = \sum_{i=1}^{n} a_i^{\mathcal{S}} f(x_i), \qquad \text{where} \qquad a_i^{\mathcal{S}} = \frac{|x - x_i|^{-p}}{\sum_{j=1}^{n} |x - x_i|^{-p}}.$$

where p is the power parameter. Therefore, it is sufficient to prove

$$\lim_{\gamma \to 0^+} a_i = a_i^{\mathcal{S}}, \quad i = 1, \ldots, n \tag{A 3}$$

with $p = 2N + 2$.

$a^{\mathcal{S}} = (a_1^{\mathcal{S}}, \ldots, a_n^{\mathcal{S}})$ with $p = 2N + 2$ is the unique solution for the constraint minimization problem

$$\mathcal{Q}^{\mathcal{S}} = \min_{\sum a_i = 1} \sum_{i=1}^{n} \left(a_i \frac{(x_i - x)^{N+1}}{(N+1)!} \right)^2 \tag{A 4}$$

Then

$$\begin{aligned} \frac{\mathcal{Q}^*(x)}{\beta^2 \gamma^{2N+2}} &= \min_{\sum a_i = 1} \frac{\mathcal{Q}(x, a_1, \ldots, a_n)}{\beta^2 \gamma^{2N+2}} \le \frac{\mathcal{Q}(x, a_1^{\mathcal{S}}, \ldots, a_n^{\mathcal{S}})}{\beta^2 \gamma^{2N+2}} \\ &= \mathcal{Q}^{\mathcal{S}} + \sum_{k=1}^{N} \gamma^{2k-2N-2} \left(\sum_{i=1}^{n} a_i^{\mathcal{S}} \frac{(x_i - x)^k}{k!} \right)^2 \ . \end{aligned}$$

The second term goes to 0 as $\gamma \to \infty$. Therefore,

$$\lim_{\gamma \to +\infty} \frac{\mathcal{Q}^*(x)}{\beta^2 \gamma^{2N+2}} \le \mathcal{Q}^{\mathcal{S}} \ , \tag{A 5}$$

On the other hand,

$$\frac{\mathcal{Q}^*(x)}{\beta^2\gamma^{2N+2}} = \frac{\mathcal{Q}(x,a_1,\ldots,a_n)}{\beta^2\gamma^{2N+2}} \ge \sum_{i=1}^n \left(a_i \frac{(x_i-x)^{N+1}}{(N+1)!} \right)^2 .$$

Therefore,

$$\lim_{\gamma\to+\infty} \sum_{i=1}^n \left(a_i \frac{(x_i-x)^{N+1}}{(N+1)!} \right)^2 \le \lim_{\gamma\to+\infty} \frac{\mathcal{Q}^*(x)}{\beta^2\gamma^{2N+2}} \le \mathcal{Q}^{\mathcal{S}} .$$

Since $\mathcal{Q}^{\mathcal{S}}$ is the minimum of the quadratic programming (A 4), the above inequality is an equality. In addition, because this quadratic programming (A 4) is non-degenerate with unique solution $a_1^{\mathcal{S}},\ldots,a_n^{\mathcal{S}}$, we conclude that Eq. (A 3) holds. □

Theorem 8. *For any interpolant function $\tilde f$ constructed with (2.1) and (2.5),*

$$\lim_{x\to\pm\infty} \tilde f(x) = \frac{1}{n}\sum_{i=1}^n f(x_i)$$

Proof. Define $a_i^\infty = \dfrac{1}{n}$. It is sufficient to prove that

$$\lim_{x\to\pm\infty} a_i = a_i^\infty, \quad i=1,\ldots,n. \tag{A 6}$$

Note that a^∞ is the solution of the constraint quadratic programming

$$\mathcal{Q}^\infty = \min_{\sum a_i=1} \sum_{i=1}^n a_i^2 \tag{A 7}$$

Then

$$\begin{aligned}\frac{(N+1)!^2}{\beta^2(\gamma x)^{2N+2}}\mathcal{Q}^*(x) &= \frac{(N+1)!^2}{\beta^2(\gamma x)^{2N+2}} \min_{\sum a_i=1} \mathcal{Q}(x,a_1,\ldots,a_n)\\ &\le \frac{(N+1)!^2}{\beta^2(\gamma x)^{2N+2}}\mathcal{Q}(x,a_1^{\mathcal{S}},\ldots,a_n^{\mathcal{S}})\\ &= \sum_{i=1}^n a_i^2 \left(\frac{x-x_i}{x}\right)^{2N+2}\\ &+ \sum_{k=1}^N \gamma^{2k-2N-2} \left(\sum_{i=1}^n a_i^{\mathcal{S}} \frac{(x_i-x)^k}{x^{N+1}} \frac{(N+1)}{!k!} \right)^2 .\end{aligned}$$

Because $\displaystyle\lim_{x\to\pm\infty} \frac{x-x_i}{x} = 1$ and $\displaystyle\lim_{x\to\pm\infty} \frac{(x_i-x)^k}{x^{N+1}} = 0$ for $i=1,\ldots,N$, the r.h.s. in the inequality above converges to $\displaystyle\sum_{i=1}^n a_i^2 = \mathcal{Q}^\infty$. Therefore,

$$\lim_{x\to\pm\infty} \frac{(N+1)!^2}{\beta^2(\gamma x)^{2N+2}}\mathcal{Q}^*(x) \le \mathcal{Q}^\infty$$

On the other hand,

$$\frac{(N+1)!^2}{\beta^2(\gamma x)^{2N+2}}\,\mathcal{Q}^*(x) = \frac{(N+1)!^2}{\beta^2(\gamma x)^{2N+2}}\,\mathcal{Q}(x, a_1, \ldots, a_n) \geq \sum_{i=1}^{n} a_i^2 \left(\frac{x - x_i}{x}\right)^{2N+2} .$$

Therefore,

$$\lim_{x\to\pm\infty} \sum_{i=1}^{n} a_i^2 = \lim_{x\to\pm\infty} \sum_{i=1}^{n} a_i^2 \left(\frac{x - x_i}{x}\right)^{2N+2} \leq \lim_{x\to\pm\infty} \frac{(N+1)!^2}{\beta^2(\gamma x)^{2N+2}}\,\mathcal{Q}^*(x) \leq \mathcal{Q}^\infty .$$

Since $\mathcal{Q}^\infty$ is the minimum of the quadratic programming (A 7), the above inequality is an equality. In addition, because this quadratic programming (A 7) is non-degenerate with unique solution $a_1^\infty, \ldots, a_n^\infty$, we conclude that Eq. (A 6) holds. □

Corollary 2. *The rational interpolant $\tilde{f}$ has no poles on the extended real line $[-\infty, +\infty]$.*

Proof. This result is obtained by combining Theorems 2 and 8. □