

Article

Quasi-Interpolation on Chebyshev Grids with Boundary Corrections

Faisal Alsharif 

Department of Mathematics, College of Science, Taibah University,
Al-Madinah Al-Munawarah 30002, Saudi Arabia; fsharif@taibahu.edu.sa

Abstract: Quasi-interpolation is a powerful tool for approximating functions using radial basis functions (RBFs) such as Gaussian kernels. This avoids solving large systems of equations as in RBF interpolation. However, quasi-interpolation with Gaussian kernels on compact intervals can have significant errors near the boundaries. This paper proposes a quasi-interpolation method with Gaussian kernels using Chebyshev points and boundary corrections to improve the approximation near the boundaries. The boundary corrections use a linear approximation of the function beyond the interval to estimate the truncation error and add correction terms. Numerical studies on test functions show that the proposed method reduces errors significantly near boundaries compared to quasi-interpolation without corrections, for both equally spaced and Chebyshev points. The convergence and accuracy with the boundary corrections are generally better with Chebyshev points compared to equally spaced points. The proposed method provides an efficient way to perform quasi-interpolation on compact intervals while controlling the boundary errors. This study introduces a novel approach to quasi-interpolation modification, which significantly enhances convergence rates and minimizes errors at boundary points, thereby advancing the methods for boundary approximation.

Keywords: quasi-interpolation; radial basis functions; boundary corrections; chebyshev points; univariate interpolation



Citation: Alsharif, F. Quasi-Interpolation on Chebyshev Grids with Boundary Corrections. *Computation* **2024**, *12*, 100. <https://doi.org/10.3390/computation12050100>

Academic Editors: Babak Shiri and Zahra Alijani

Received: 31 March 2024
Revised: 5 May 2024
Accepted: 7 May 2024
Published: 13 May 2024



Copyright: © 2024 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Let f be a d -variate real valued function defined on the bounded domain $\Omega \subset \mathbb{R}^d$. There is an extensive research in the literature to interpolate/approximate such a function f over an equally spaced grid of points using radial basis functions (RBF) with a high degree of accuracy [1,2]. As [1] argued that radial basis functions overcome the problems of using high dimensional polynomials to attain the specified degree of smoothness for the interpolants, they became very popular among the applied scientists for use in applied problems [3–6].

Suppose, the data on the function values. $f(x_i)$, are available for a set of points x_i , $i = 1, \dots, N$. Then a radial basis function (RBF) interpolant can be defined as follows

$$If(x) = \sum_{j=1}^N c_j \phi(r_j) \quad (1)$$

where r_j is the Euclidean distance between the point x and the grid point x_j with $r_j = \|x - x_j\|$ and c_j are the constants which are determined from the constraints

$$f(x_i) = If(x_i), \quad i = 1, \dots, N \quad (2)$$

One of the most popular choice for the radial basis function is the Gaussian kernel for ϕ , that is,

$$\phi(x) = \phi_h(x) = \frac{1}{\sqrt{2\pi}h} e^{-x^2/(2h^2)} \quad (3)$$

where h is a scale parameter which determines the degree of smoothness in the interpolant. The convergence properties of the RBF interpolation with the Gaussian kernel depend on the distances between the grid points x_i and the parameter h for the kernel.

Users find it very attractive due to its simplicity in definition and high-degree of accuracy. However, to obtain the constants c_j , one needs to solve a system of N linear equations and quite often the system is ill-conditioned, resulting in unstable solutions. The ill-conditioning of the system of equations with the Gaussian kernel depends on the shape parameter h as define above. Though there are a number of modifications proposed in the literature to overcome the ill-conditioning of the systems [7,8], all such proposals are computationally complex and that restricts their use only on a equally spaced grid for functions defined on lower-dimensional domains.

One particular and powerful tool for tackling multidimensional problems is quasi-interpolation [9,10], which avoids solving a large system of ill-conditioned linear equations and is very popularly used in science and engineering. A major advantage of quasi-interpolation is that the interpolant retains the smoothness of the kernel function being used. The quasi-interpolation with Gaussian kernels has simplicity in its definition and has good shape properties (convex, concave, or linear depending on the original function). The interpolant also retains the property of exponential decay at infinity of the Gaussian kernel. As quasi-interpolation provides direct solutions without any need to solve large algebraic systems of equations [11], when compared to other mesh-free techniques such as RBF interpolation, this approach can approximate the function in less computational time in higher dimensions. Quasi-interpolation has been successfully applied to scattered data approximation and interpolation, numerical solutions to partial differential equations, and quadrature.

Maz'ya and Schmidt [12] discussed the convergence rates of some of the quasi-interpolation methods commonly in use. Even though quasi-interpolation operators have generally used functions defined on \mathbb{R} , there are some applications that require functions to be defined on a compact interval to allow efficient approximation, such as boundary integral equations and treatment of partial differential equations. Müller and Varnhorn [13] applied the quasi interpolation method to functions with Hölder continuous derivatives and established convergence rates for the quasi-interpolants. In contrast to functions defined on all real numbers, \mathbb{R} , a truncation error has to be controlled for the functions, which are defined on a compact subset. This article starts with an explanation of why Chebyshev grids exist and provides numerical study. Followed by the derivation of boundary corrections to describe how to derive them with rigorous theoretical justification and apply boundary corrections. Then, the numerical experiments section provides extensive qualitative and quantitative evidence of the benefits provided by the boundary corrections, including computations and error estimation. In the last section, it concluded with a summary of this article. The novelty of our study lies in the introduction of a pioneering quasi-interpolation modification technique, aimed at improving convergence behaviour and reducing boundary errors.

2. Quasi-Interpolation with Gaussian Kernels

The main motivation for the quasi-interpolation is as follows. Consider the convolution of the univariate function f with the Gaussian kernel ϕ ,

$$Lf(x) = \int_{-\infty}^{\infty} f(t)h^{-1}\phi((x-t)/h)dt \quad (4)$$

where

$$\phi(x) = \frac{1}{\sqrt{2\pi}}e^{-x^2/2}$$

is the probability density function (p.d.f.) of the standard Gaussian distribution with mean 0 and standard deviation 1. Cheney [14] showed that the convolution $Lf(x)$ converges to f as $h \rightarrow 0$, with the convergence being uniform on compact sets [14]. The rate of convergence

depends upon the smoothness of the function f . Now, the convolution operator can be approximated by its discrete counter-part:

$$Q_h f(x) = \sum_{i \in \mathbb{Z}} f(ih) \frac{1}{h} \phi\left(\frac{(x - ih)}{h}\right). \tag{5}$$

Due to this representation of the convolution operator, one can establish the convergence properties of the quasi-interpolant $Q_h f$ on a compact subset of \mathbb{R} with equally spaced points on \mathbb{R} . For practical purposes, one can consider without loss of generality the compact set $[-1, 1]$ as the support for the function f , where one wishes to study the interpolation. Also note that, $Q_h f$ is defined on an infinite grid. For practical implementation, we define

$$\hat{Q}_h f(x) = \sum_{j=1}^N f(x_j) \phi\left(\frac{x - x_j}{h}\right), \tag{6}$$

where x_1, x_2, \dots, x_N are equally spaced points in the interval $[-1, 1]$ and $h = x_j - x_{j-1}$. As the number of points $N \rightarrow \infty$, the distance between the points $h \rightarrow 0$ and convergence results hold. However, it is easy to observe that the discrete convolution operator $\hat{Q}_h f$ only approximates the integral

$$\int_{-1}^1 f(t) h^{-1} \phi((t - x)/h) dt. \tag{7}$$

If the function f is defined beyond the in the interval $[-1, 1]$, the approximation by this quasi-interpolation is not good around the boundary of -1 and 1 .

In this article, we will study the error due to this truncation at the boundary and will present numerical examples. In Section 2, we will present the some mathematical approach to correct the approximation near the boundary. We will also study the performance of our boundary correction for some numerical examples.

Chebyshev Points

The points

$$x_k = \cos\left(\frac{(2k - 1)\pi}{2N}\right), \tag{8}$$

for $k = 1, \dots, N$ are called Chebyshev points. They are the roots of the degree N Chebyshev polynomial defined by

$$T_N(x) = \cos(N \arccos x),$$

for $x \in [-1, 1]$. The Chebyshev polynomials satisfy the recursion formula $T_0(x) = 1$, $T_1(x) = x, \dots, T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x)$, for $k \geq 1$, and thus the leading coefficient of T_k is 2^{k-1} . Moreover, observe that

$$|T_N(x)| = |\cos(N \arccos(x))| \leq 1$$

for $x \in [-1, 1]$. Thus, if x_1, \dots, x_N are Chebyshev points, then

$$|(x - x_1) \cdots (x - x_N)| = \left| \frac{1}{2^{N-1}} T_N(x) \right| \leq \left| \frac{1}{2^{N-1}} \right|,$$

for $x \in [-1, 1]$.

Using this argument, Ref. [15] showed that a polynomial approximation to a real-valued continuous function f defined on $[-1, 1]$ can have a convergence rate at the order of 2^{N-1} if the approximation is evaluated at the Chebyshev points which is a lot better than the convergence rates $O(N^{-1})$ achieved by the equally spaced points.

The faster convergence rates with polynomial interpolation evaluated at the Chebyshev points is our main motivation for using Chebyshev points in the quasi interpolation problems under consideration. In this research paper, we will investigate the Chebyshev

points as an alternative to the equally spaced grid points to evaluate the performance and convergence results numerically of our proposed quasi-interpolation algorithm with Gaussian kernel and boundary corrections. The modified quasi-interpolation with Chebyshev points can be formulated as

$$\hat{Q}_h f(x) = \sum_{j=1}^N \frac{|x_j - x_{j-1}|}{h} f(x_j) \phi\left(\frac{x - x_j}{h}\right), \tag{9}$$

where x_1, x_2, \dots, x_N are the Chebyshev points in the interval $[-1, 1]$.

In the absence of established theoretical results on the convergence rates for the quasi-interpolation with Gaussian kernels evaluated at the Chebyshev points, we will study numerically with different examples.

3. Boundary Corrections Formula

We have mentioned earlier that the convolution operator $Lf(x)$ converges to $f(x)$ as $h \rightarrow 0$ in the Gaussian kernel. However, the discrete quasi interpolant $\hat{Q}_h f(x)$ approximates the integral

$$\int_{-1}^1 f(t) \frac{1}{h} \phi\left(\frac{t - x}{h}\right) dt \tag{10}$$

as $N \rightarrow \infty$. We can now write the convolution as:

$$\begin{aligned} Lf(x) &= \int_{-\infty}^{\infty} f(t) h^{-1} \phi((t - x)/h) dt \\ &= \int_{-\infty}^{-1} f(t) \frac{1}{\sqrt{2\pi}h} e^{-(t-x)^2/(2h^2)} dt + \int_{-1}^1 f(t) \frac{1}{\sqrt{2\pi}h} e^{-(t-x)^2/(2h^2)} dt \\ &\quad + \int_1^{\infty} f(t) \frac{1}{\sqrt{2\pi}h} e^{-(t-x)^2/(2h^2)} dt \\ &= I_1 + I_2 + I_3 \end{aligned}$$

where

$$I_1 = \int_{-\infty}^{-1} f(t) \frac{1}{\sqrt{2\pi}h} e^{-(t-x)^2/(2h^2)} dt,$$

$$I_2 = \int_{-1}^1 f(t) \frac{1}{\sqrt{2\pi}h} e^{-(t-x)^2/(2h^2)} dt,$$

and

$$I_3 = \int_1^{\infty} f(t) \frac{1}{\sqrt{2\pi}h} e^{-(t-x)^2/(2h^2)} dt$$

We have noted that $\hat{Q}_h f(x)$ approximates the integral I_2 . If x is much closer to the boundary, the integrand in I_1 will be small and I_1 may be negligible. Similarly, if x is far away from the boundary, the integrand in I_3 will be small and I_3 might be negligible. Therefore, if the point of interpolation x is well within the interval $[-1, 1]$, both the integrals I_1 and I_3 will negligible and the approximation of I_2 using the quasi-interpolant $\hat{Q}_h f(x)$ will be a good approximation of the convolution $Lf(x)$ and in turn that will converge to $f(x)$ as $h \rightarrow 0$ and $N \rightarrow \infty$. However, if the point of interpolation x is close to the boundary of -1 or 1 , the integrals I_1 and I_3 are not negligible.

To improve the approximation of $f(x)$, when x is close to the boundary, we need to have some approximation to the integrals I_1 and I_3 . as we do not know the form of $f(x)$ outside the interval $[-1, 1]$, we approximate it with two linear functions which are continuous and differentiable at $x = -1$ and $x = 1$. Let us assume,

$$f(x) = a_1(x + 1) + b_1, \quad \text{for } x \leq -1$$

$$f(x) = a_2(x - 1) + b_2, \quad \text{for } x \geq 1.$$

From continuity, we must have

$$b_1 = f(-1), \quad b_2 = f(1)$$

and

$$a_1 = f'(-1), \quad a_2 = f'(1).$$

Using these approximations, one write I_1 as

$$\begin{aligned} I_1 &= \int_{-\infty}^{-1} (a_1(t+1) + b_1) \frac{1}{h\sqrt{2\pi}} e^{-(t-x)^2/(2h^2)} dt \\ &= a_1 \int_{-\infty}^{-1} (t-x) \frac{1}{h\sqrt{2\pi}} e^{-(t-x)^2/(2h^2)} dt \\ &\quad + (a_1(x+1) + b_1) \int_{-\infty}^{-1} \frac{1}{h\sqrt{2\pi}} e^{-(t-x)^2/(2h^2)} dt \\ &= a_1 I_{11} + (a_1(x+1) + b_1) I_{12} \end{aligned}$$

Now,

$$\begin{aligned} I_{11} &= \int_{-\infty}^{-1} (t-x) \frac{1}{h\sqrt{2\pi}} e^{-(t-x)^2/(2h^2)} dt \\ &= \int_{-\infty}^{-(x+1)/h} uh \frac{1}{\sqrt{2\pi}} e^{-u^2/2} du \\ &= -\frac{h}{\sqrt{2\pi}} e^{-(x+1)^2/(2h^2)} = -h \cdot \phi\left(\frac{x+1}{h}\right) \end{aligned}$$

where

$$\phi(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}.$$

We also simplify,

$$\begin{aligned} I_{12} &= \int_{-\infty}^{-1} \frac{1}{h\sqrt{2\pi}} e^{-(t-x)^2/(2h^2)} dt \\ &= \int_{-\infty}^{-(x+1)/h} \frac{1}{\sqrt{2\pi}} e^{-u^2/2} du = \Phi\left(-\frac{(x+1)}{h}\right) \end{aligned}$$

where $\Phi(x)$ is the distribution function of the standard Gaussian distribution

$$\Phi(x) = \int_{-\infty}^x \phi(u) du.$$

Therefore, with the linear approximation of the function $f(x)$, the integral I_1 can be approximated by

$$I_1 \approx -a_1 h \phi\left(\frac{x+1}{h}\right) + (a_1(x+1) + b_1) \Phi\left(-\frac{(x+1)}{h}\right).$$

Similarly, the integral I_3 can be approximated by

$$I_3 \approx a_2 h \phi\left(\frac{1-x}{h}\right) + (a_2(x-1) + b_2) \left[1 - \Phi\left(\frac{(1-x)}{h}\right)\right].$$

Combining all of them together, we propose the quasi-interpolation function with boundary corrections as follows:

$$\begin{aligned} \tilde{Q}_h f(x) &= -a_1 h \phi\left(\frac{x+1}{h}\right) + (a_1(x+1) + b_1) \Phi\left(-\frac{(x+1)}{h}\right) \\ &\quad \hat{Q}_h f(x) + a_2 h \phi\left(\frac{1-x}{h}\right) + (a_2(x-1) + b_2) \left[1 - \Phi\left(\frac{(1-x)}{h}\right)\right]. \end{aligned}$$

Now, note that as $h \rightarrow 0$, $I_1 \rightarrow 0$ and $I_3 \rightarrow 0$ for all $x \in (-1, 1)$. For $x = -1$, $I_1 \rightarrow b_1/2$ and for $x = 1$, $I_3 \rightarrow b_2/2$ as $h \rightarrow 0$. Therefore, the convergence of the quasi-interpolation function $\tilde{Q}_h f(x)$ at $x = -1$ and $x = 1$ needs to be investigated in more detail and it may be possible that this interpolation may not converge at these two points.

4. Numerical Experiments

In this section, we investigate the performance of our proposed quasi-interpolation with boundary corrections, $\tilde{Q}_h f(x)$ with equally spaced grid points as well as Chebyshev points. For this purpose, we consider the following 4 test functions, which are widely used in the literature [6,16,17]:

- $f_1(x) = \sin(4\pi x)$.
- $f_2(x) = \cosh(x) \exp(\sinh(x))$.
- $f_3(x) = \exp\left(\frac{1}{1+x^2}\right) \tanh(x/10\pi)/(1 + 16x^3)$.
- $f_4(x) = -2xe^{-x} \cos(20x)$.

The performance of quasi-interpolation with and without boundary corrections were evaluated in a grid of 100 points in the interval $[-1, 1]$. As the metric for performance, we compute maximum absolute error and the root mean-square error as defined below:

$$\max \text{ error} = \max_i |f(x_i) - \tilde{Q}_h f(x_i)| \tag{11}$$

$$\text{RMS Error} = \sqrt{\frac{1}{n} \sum_{i=1}^n (f(x_i) - \tilde{Q}_h f(x_i))^2} \tag{12}$$

where $n = 100$ and x_i 's are the grid points at which the function is approximated.

In Figure 1, we illustrate the approximation using $N = 100$ equally spaced grid points. Observe that the error of the interpolated function is considerably large at the boundary points of the interval $[-1, 1]$, while the errors for interior points are small when no boundary corrections were used. On the other hand, the errors are very similar for all points, when the boundary corrections are used.

Similar to the equally-spaced grid points for interpolations, the results for interpolation with Chebyshev points are shown in Figure 2 for the function $f_1(x)$. We again observe that the error are higher at the boundaries $x = -1$ and $x = 1$ when there are no boundary corrections and the errors reduce with the boundary corrections.

In Table 1, we present the absolute maximum error and RMS error for the quasi-interpolation of the function $f_1(x)$ with and without boundary corrections. We observe that the interpolation results converge nicely as N increases for both equally spaced points and Chebyshev points. The errors for the interpolation with boundary corrections are always smaller than those without the boundary corrections for higher values of N .

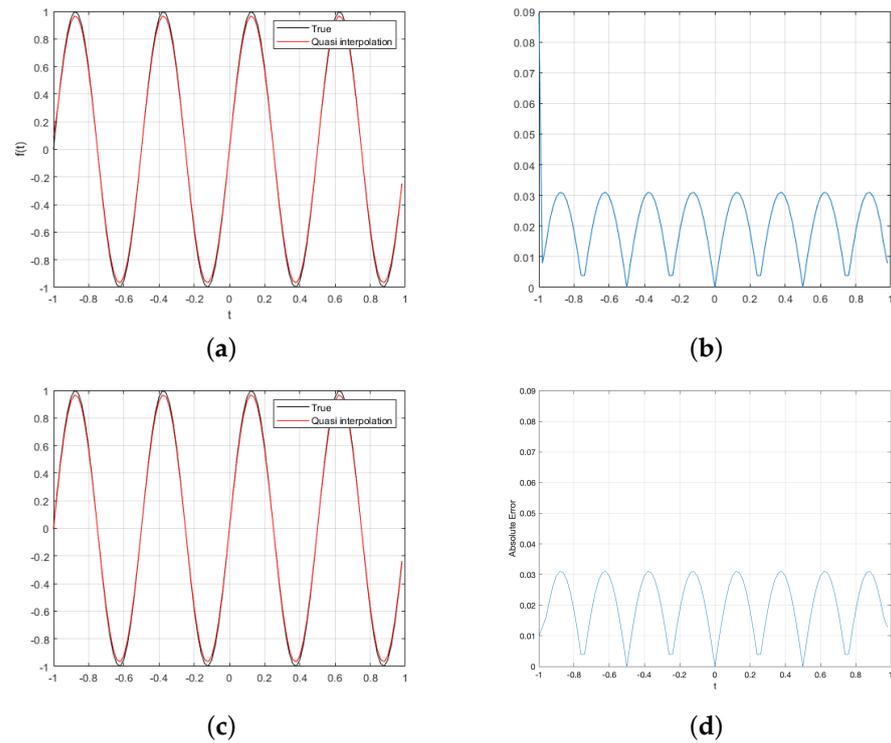


Figure 1. The function approximation for $f_1(x)$ and the absolute errors with $N = 100$ equally spaced grid points. (a) Interpolation without boundary correction, (b) Errors without boundary correction, (c) Interpolation with boundary correction, and (d) Errors with boundary correction.

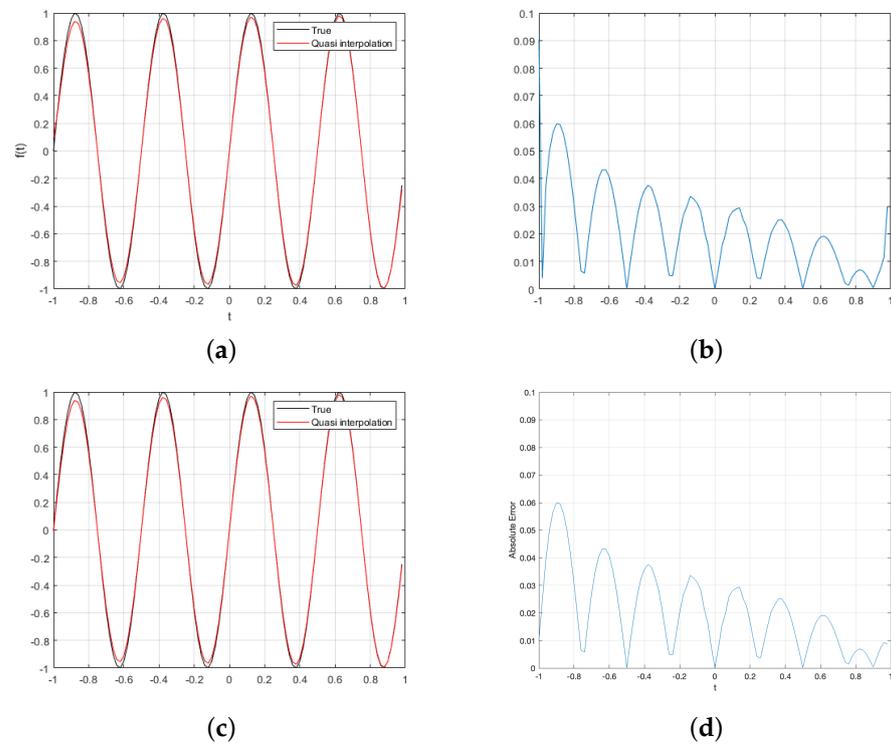


Figure 2. The function approximation for $f_1(x)$ and the absolute errors with $N = 100$ Chebyshev points. (a) Interpolation without boundary correction, (b) Errors without boundary correction, (c) Interpolation with boundary correction, and (d) Errors with boundary correction.

Table 1. Performance metrics with maximum absolute error and RMS error for the quasi-interpolation with equally spaced points and Chebyshev points with and without boundary corrections for the function $f_1(x)$.

N	Equal				Chebyshev			
	Without Correction		With Correction		Without Correction		With Correction	
	Max Error	RMS Error	Max Error	RMS Error	Max Error	RMS Error	Max Error	RMS Error
10	0.9583	0.4448	1.0025	0.4734	1.2046	0.4529	1.3020	0.5961
10^2	0.0893	5.6×10^{-4}	0.0310	4.9×10^{-4}	0.0903	7.1×10^{-4}	0.0598	6.3×10^{-4}
10^3	0.0091	8.9×10^{-7}	8.9×10^{-4}	5.7×10^{-8}	0.0098	2.6×10^{-6}	0.0034	1.6×10^{-6}
10^4	9.1×10^{-4}	8.3×10^{-9}	8.8×10^{-5}	8.3×10^{-11}	9.9×10^{-4}	5.3×10^{-8}	5.2×10^{-4}	4.4×10^{-8}

The approximations to the function $f_2(x)$ with $N = 10$ equally spaced grid points are presented in Figure 3. We observe that when there is no boundary correction, the quasi-interpolation approximates the function well for points which are smaller than 0.9. However, the approximation is poor at the edges. But when the boundary corrections are used, the quasi-interpolation picked up the correct shape of the even at the edges, though the approximation is not very good with $N = 10$ points only. The errors are also increasing nearer the upper boundary of the interval $[-1, 1]$ even with boundary corrections.

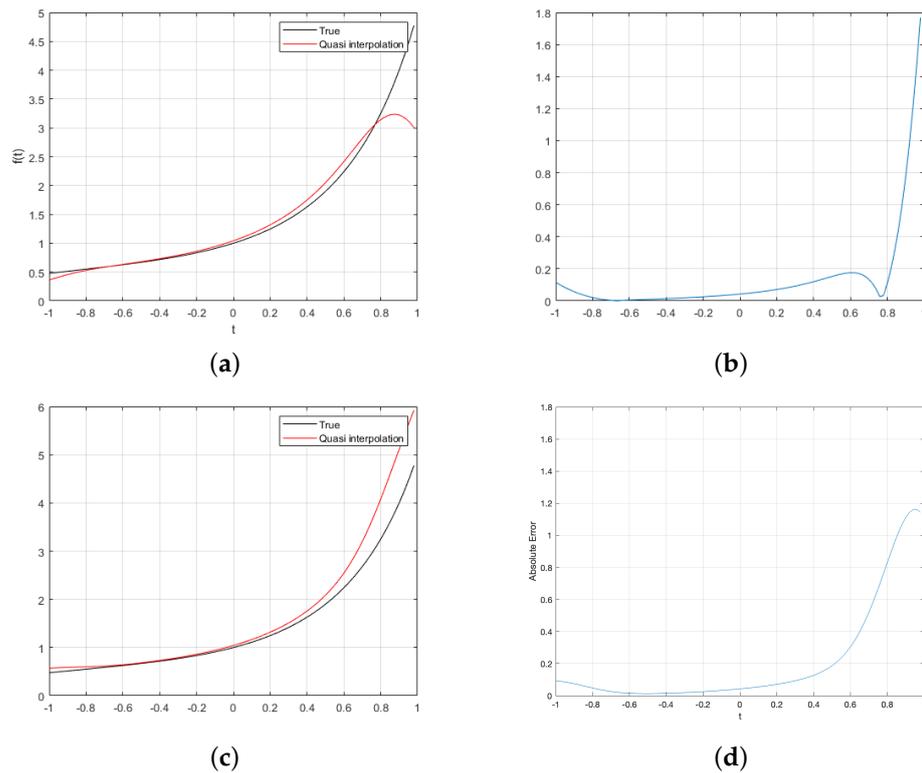


Figure 3. The function approximation for $f_2(x)$ and the absolute errors with $N = 10$ equally spaced grid points. (a) Interpolation without boundary correction, (b) Errors without boundary correction, (c) Interpolation with boundary correction, and (d) Errors with boundary correction.

Similar approximations to $f_2(x)$ with $N = 10$ Chebyshev points are made in Figure 4. We observe almost a similar pattern with the errors as observed for the case with equally spaced points.

Detailed performance metrics for varying values of grid points n are presented in Table 2 for the function $f_2(x)$. We observe that the approximations do not converge so well for the equally spaced points with or without corrections. However, the errors become

smaller with the increase in N for Chebyshev points. In this example, the performance of the Chebyshev points are better than the equally spaced points. In both cases, the error with boundary corrections are small that the interpolation without boundary corrections.

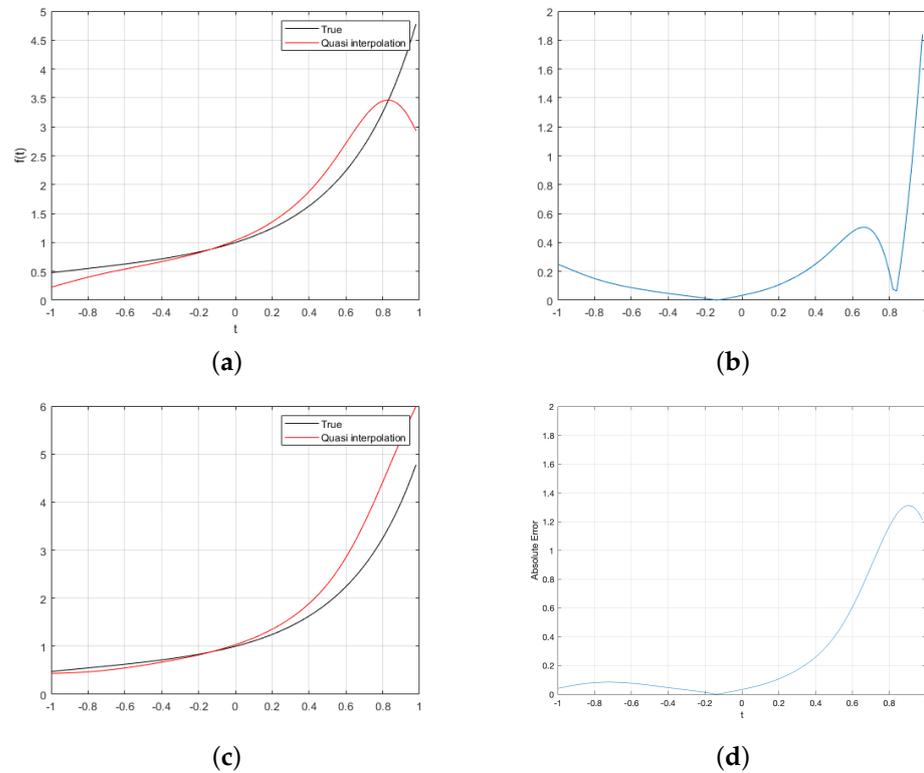


Figure 4. The function approximation for $f_2(x)$ and the absolute errors with $N = 10$ Chebyshev grid points. (a) Interpolation without boundary correction, (b) Errors without boundary correction, (c) Interpolation with boundary correction, and (d) Errors with boundary correction.

Table 2. Performance metrics with maximum absolute error and RMS error for the quasi-interpolation with equally spaced points and Chebyshev points with and without boundary corrections for the function $f_2(x)$.

N	Equal				Chebyshev			
	Without Correction		With Correction		Without Correction		With Correction	
	Max Error	RMS Error	Max Error	RMS Error	Max Error	RMS Error	Max Error	RMS Error
10	1.7683	0.0964	1.1617	0.1453	1.8394	0.1290	1.3123	0.2567
10^2	0.3013	0.0011	0.5102	0.0028	0.4218	0.0045	0.3903	0.0040
10^3	0.1429	2.0×10^{-4}	0.0950	9.0×10^{-5}	0.2475	6.5×10^{-4}	0.0371	3.53×10^{-5}
10^4	0.1432	2.0×10^{-4}	0.0950	9.0×10^{-5}	0.2413	5.8×10^{-4}	0.0037	5.0×10^{-7}

For $N = 10$ equally spaced points, the quasi-interpolation approximations evaluated at 100 points in the interval $[-1, 1]$ are presented in Figure 5 with and without boundary corrections for the function $f_3(x)$. Observe that the function f_3 is almost 0 at all points except for a spike at -0.4 . As the function is nearly 0 at boundary, the approximations with or without corrections do not differ much. For $N = 10$ points only, the peak at $x = -0.4$ is missed but it still approximates with a smooth peak at that point instead of a sharp peak.

Figure 6 shows the approximation to the function with $N = 10$ Chebyshev points. Similarly, it also misses out the peak at $x = -0.4$. This approximation is almost constant at all points in $[-1, 1]$. With Chebyshev points also, we do not see any substantial difference in quasi-interpolation approximation with or without boundary corrections.

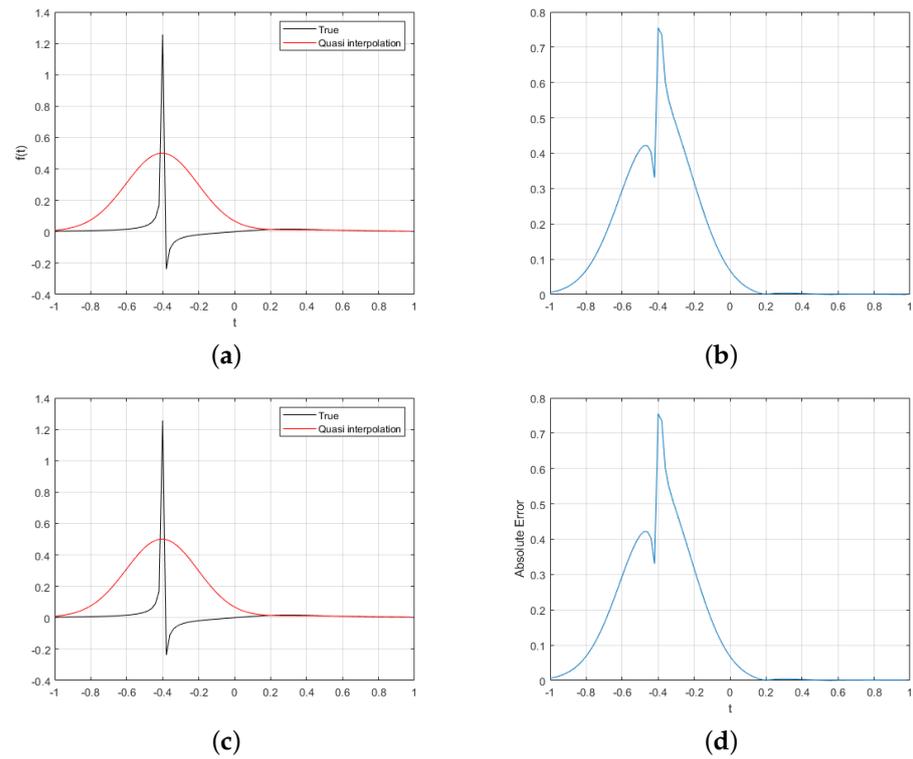


Figure 5. The function approximation for $f_3(x)$ and the absolute errors with $N = 10$ equally spaced grid points. (a) Interpolation without boundary correction, (b) Errors without boundary correction, (c) Interpolation with boundary correction, and (d) Errors with boundary correction.

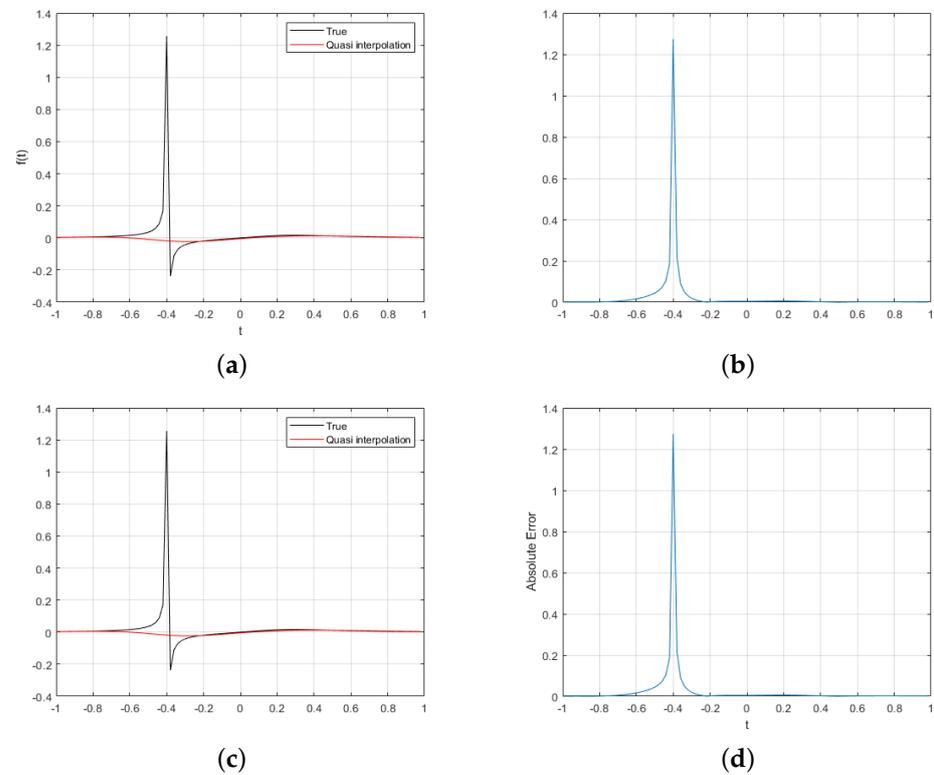


Figure 6. The function approximation for $f_3(x)$ and the absolute errors with $N = 10$ Chebyshev grid points. (a) Interpolation without boundary correction, (b) Errors without boundary correction, (c) Interpolation with boundary correction, and (d) Errors with boundary correction.

From Table 3, we observe that the convergence of the quasi-interpolation approximations with equally spaced points is slower with very small errors are attained only for large values of N . the performance of the Chebyshev points is even worse than equally spaced points for small N . However, Chebyshev points also yield similar small errors for very large values of N .

Table 3. Performance metrics with maximum absolute error and RMS error for the quasi-interpolation with equally spaced points and Chebyshev points with and without boundary corrections for the function $f_3(x)$.

N	Equal				Chebyshev			
	Without Correction		With Correction		Without Correction		With Correction	
	Max Error	RMS Error	Max Error	RMS Error	Max Error	RMS Error	Max Error	RMS Error
10	0.7553	0.0503	0.7553	0.0503	1.2743	0.0174	1.2743	0.0174
10^2	0.7729	0.0083	0.7729	0.0083	6.2833	0.8648	6.2833	0.8648
10^3	0.0380	1.5×10^{-5}	0.0380	1.5×10^{-5}	2.3462	0.0550	2.3462	0.0550
10^4	0.0051	2.7×10^{-7}	0.0051	2.7×10^{-7}	0.0052	3.0×10^{-7}	0.0052	2.7×10^{-7}

Finally, Figure 7 shows the quasi-interpolation approximations for the function $f_4(x)$ with $N = 100$ equally spaced points for interpolation. We observe that the approximations are not too bad for the points x away from the left boundary -1 . However, at the points closed to the boundary $x = -1$, the errors in approximation are large, with boundary corrections, the errors reduce a little but still there are large error near the boundary.

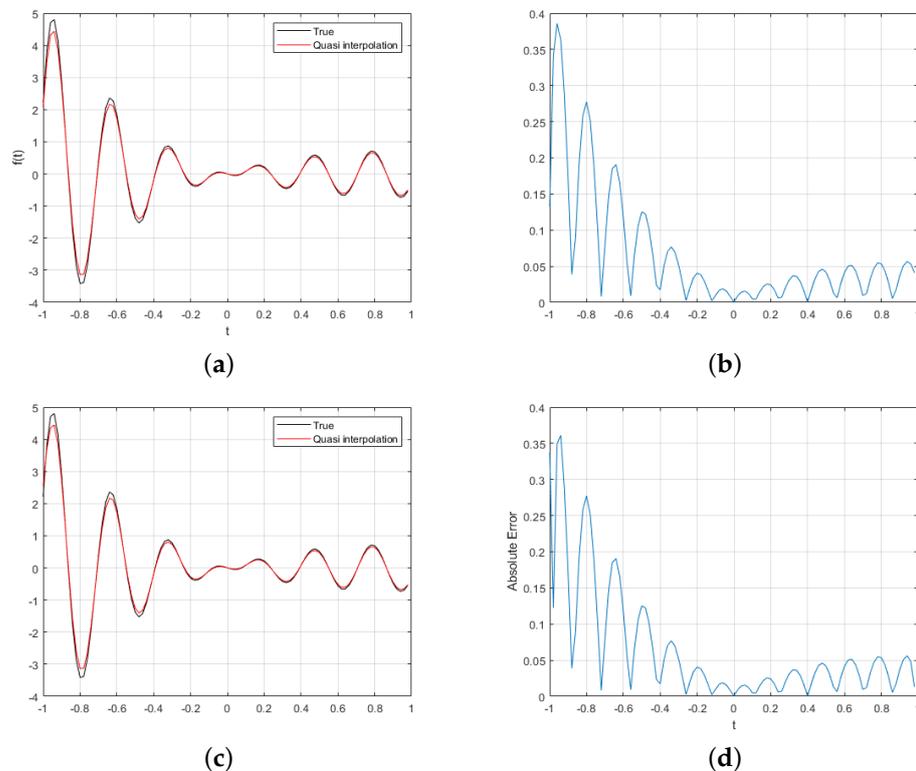


Figure 7. The function approximation for $f_4(x)$ and the absolute errors with $N = 100$ equally spaced grid points. (a) Interpolation without boundary correction, (b) Errors without boundary correction, (c) Interpolation with boundary correction, and (d) Errors with boundary correction.

The quasi-interpolation approximations of $f_4(x)$ using $N = 100$ Chebyshev points are shown in Figure 8. It also shows a similar pattern as with equally spaced points. the boundary correction did not produce any significant improvement for this problem.

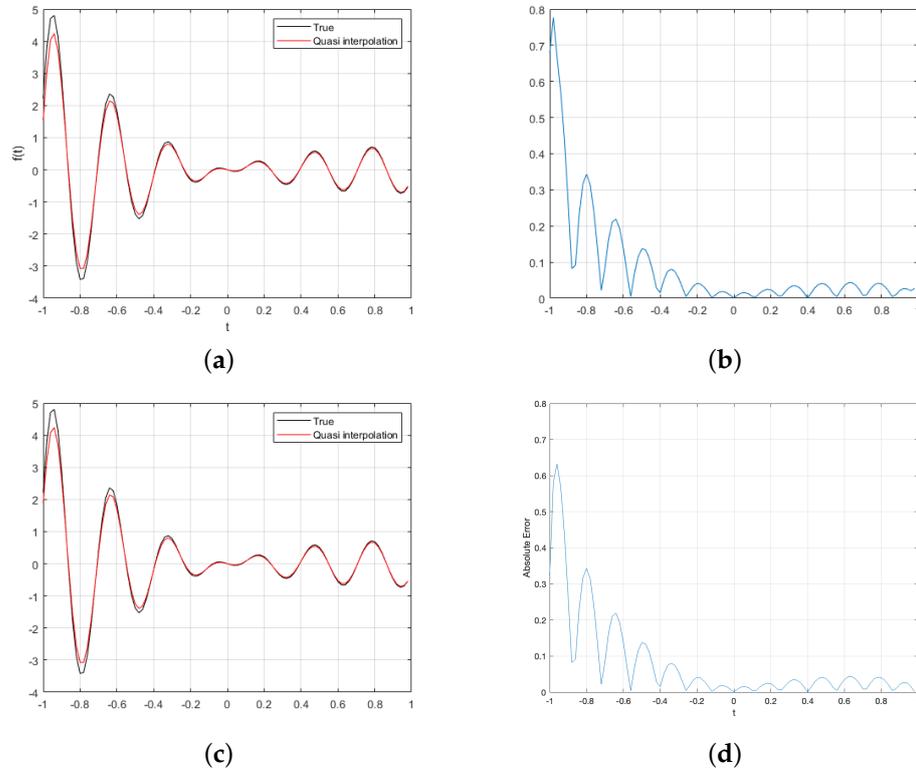


Figure 8. The function approximation for $f_4(x)$ and the absolute errors with $N = 100$ Chebyshev points. (a) Interpolation without boundary correction, (b) Errors without boundary correction, (c) Interpolation with boundary correction, and (d) Errors with boundary correction.

The detailed errors reported in Table 4 shows that the interpolation without boundary corrections did not improve much even with large number of grid points N with equally space points. We have a slightly better approximation with boundary corrections for equally space points. However, the real improvement with boundary corrections is noticeable with Chebyshev points. In this case, we have a very good convergence rate with boundary corrections and poor performance when we do not use a boundary correction. The calculations detailed in the article were performed using MATLAB 2021b within a Windows 10 environment, leveraging the processing power of an Intel i7 8th generation core.

Table 4. Performance metrics with maximum absolute error and RMS error for the quasi-interpolation with equally spaced points and Chebyshev points with and without boundary corrections for the function $f_4(x)$.

N	Equal				Chebyshev			
	Without Correction		With Correction		Without Correction		With Correction	
	Max Error	RMS Error	Max Error	RMS Error	Max Error	RMS Error	Max Error	RMS Error
10	4.8540	1.7682	3.8431	1.2506	4.7270	1.8765	6.9656	3.1055
10^2	0.3860	0.0111	0.3612	0.0107	0.7711	0.0288	0.6329	0.0222
10^3	0.5991	0.0036	0.4356	0.0019	1.0812	0.0117	0.0477	5.9×10^{-5}
10^4	0.6599	0.0044	0.4419	0.0020	1.1164	0.0125	0.0148	2.5×10^{-6}

The execution times in seconds for the proposed quasi-interpolations algorithms with and without boundary corrections are given in Table 5. All time are reported for interpolation on a grid of 100 points in the interval $[-1, 1]$. We observe that the execution times are quite small even when the number of points used in the interpolation N is as high as 10,000. The execution times for the quasi-interpolation with boundary corrections are higher than those without boundary corrections, but the time required for interpolation with equally spaced points are almost similar with the time required for interpolation with the Chebyshev points.

Table 5. Execution times (in seconds) for the quasi-interpolation with equally spaced points and Chebyshev points with and without boundary corrections.

Function	N	Equal		Chebyshev	
		Without Correction	With Correction	Without Correction	With Correction
f_1	10	0.011239	0.014522	0.010205	0.016102
	100	0.011023	0.016676	0.009124	0.016008
	1000	0.010513	0.019334	0.010005	0.017596
	10,000	0.015518	0.025141	0.015375	0.021826
f_2	10	0.009056	0.015789	0.009593	0.01635
	100	0.009874	0.016175	0.008597	0.015441
	1000	0.009885	0.018126	0.010193	0.018209
	10,000	0.015531	0.023131	0.016509	0.022192
f_3	10	0.008346	0.016715	0.009125	0.01613
	100	0.011124	0.016526	0.00895	0.01632
	1000	0.0128	0.017859	0.012898	0.018343
	10,000	0.016643	0.022995	0.016581	0.023208
f_4	10	0.012326	0.015656	0.01014	0.015537
	100	0.00898	0.016466	0.009816	0.017255
	1000	0.010226	0.017338	0.01101	0.017479
	10,000	0.015796	0.023332	0.016352	0.021948

5. Discussion and Funding

In this article, we have proposed a quasi-interpolation method with Gaussian kernels using Chebyshev points and boundary corrections. We noted that the usual quasi-interpolation approximates the function quite well for the points well within the compact interval of estimation, However, they do not approximate well when the interpolation points are close to the boundary. As a remedial measure, we have considered linear approximations of the function f beyond our close interval $[-1, 1]$, which is continuous and differentiable at the boundary points of $x = -1$ and $x = 1$. Using that linear shape of the function f , we propose some corrections terms go to 0 if the point x is well within the interval $[-1, 1]$ and $h \rightarrow 0$. Therefore, the approximation with or without boundary corrections should work similarly for points within the interval $[-1, 1]$. However, the boundary corrections contribute significantly, if x is close to the boundary -1 or 1 and the function f is not close to 0 at those points.

In our numerical studies, we have shown that for all functions, the Chebyshev points for interpolation produce reasonable convergence results with or without boundary corrections. The function $f_1(x) = \sin(4\pi x)$ is a periodic function with equal periods and amplitudes. In this example, the boundary corrections with Chebyshev points provides a little improvement over the interpolation without corrections, but the performance for equally spaced points are better than the Chebyshev points.

For the exponentially increasing function $f_2(x) = \cosh(x) \exp(\sinh(x))$, the approximations are not too bad near the lower boundary of -1 as the value of f_2 nears 0 at that boundary, but as the function increases steeply at the upper boundary $x = 1$, the quasi interpolation without boundary correction completely misses the increasing pattern near

the boundary and starts decreasing with small values of N . With boundary corrections, that problem at $x = 1$ is no longer there and also the approximation at the lower end point $x = -1$ is even better. For smaller values of N , equally spaced points provide a better approximation than the Chebyshev points, but with large values of N Chebyshev points with boundary corrections provide the the best approximations. Thus, for such functions, Chebyshev points has a better convergence rate.

The function $f_3(x) = \exp\left(\frac{1}{1+x^2}\right) \tanh(x/10\pi)/(1 + 16x^3)$ is constant for most part of the interval $[-1, 1]$ with a sudden peak at $x = -0.4$. Since this function is almost zero near the point $x = -0.4, -1$ and 1 . The boundary corrections do not have any significant effect on the approximation. With small values of N , the interpolation points missed out the peak and had a very poor approximation. However, as N increases substantially to 10,000, both equally spaced points and Chebyshev points based interpolation performs similarly.

Our last function $f_4(x) = -2xe^{-x} \cos(20x)$ has oscillatory behaviour with exponential damping. the magnitude of the function is maximum at the lower end point $x = -1$ and it oscillates with a very small amplitude at the upper boundary of $x = 1$. For this behaviour, the maximum errors in approximations are observed near the lower boundary. In this example, there is almost no convergence or very slow convergence when there are no boundary corrections. With boundary corrections and Chebyshev points we observe a dramatic improvement in approximation.

Finally, we observed from our numerical studies that boundary corrections only play a significant role when the function f is away from 0 near the boundary. In this study, we have approximated the function f beyond the compact interval of $[-1, 1]$ using linear functions. However, the approximation may not perform well if the function is very different from linear around the boundary. To improve one can derive similar boundary correction terms for quadratic or polynomial approximations. Another problem to note that our approximation uses the same direction of the function at the boundary, that is, if it is increasing at -1 , it remain increasing linearly at that point and if that is decreasing, it is decreasing at that point. Due to the nature of approximation, it does not matter if the function changes the direction at a point far way from the boundary. But if the the boundary $x = -1$ or $x = 1$ are the stationary points or points of inflection, then boundary corrections may bring in more errors in approximations.

In this study, we have applied quasi-interpolation with boundary corrections only for univariate functions, however, one can extend the boundary corrections to higher dimensions as well though that will require some mathematically intensive derivation of the integrals at the boundaries and there is no easy simplifications. As a future extension of this study, one can explore that.

6. Conclusions

This paper presented a quasi-interpolation method using Gaussian kernels with Chebyshev points and boundary corrections to improve approximation accuracy near the boundaries for functions defined on compact intervals.

The proposed boundary corrections estimate the truncation error at the boundaries by approximating the function's continuation beyond the interval using linear functions. This allows adding correction terms to the standard quasi-interpolation formula.

Numerical studies on test functions showed that the boundary corrections significantly reduced errors near the boundaries compared to quasi-interpolation without corrections. The convergence and accuracy with the corrections were generally better using Chebyshev points instead of equally spaced points.

The results demonstrate that the proposed approach provides an efficient and accurate way to perform quasi-interpolation for functions on compact intervals. It overcomes the issue of large boundary errors in standard quasi-interpolation.

The application of quasi-interpolation holds significant promise across a spectrum of disciplines. In physics, it can facilitate accurate modelling in the simulation of particle interactions, fluid dynamics, and electromagnetic fields. In mechanics, quasi-interpolation

techniques can enhance finite element analysis by providing more precise representations of structural behaviour under varying loads and boundary conditions.

7. Recommendation for Future Research

Studying theoretical convergence rates with the proposed boundary corrections Extending the boundary corrections approach to higher dimensions Using polynomial or spline approximations beyond boundaries instead of linear Applying the method to solve PDEs and integral equations on compact domains Overall, this work enhances quasi-interpolation with Gaussian kernels for problems requiring function approximation on compact intervals with high accuracy near the boundaries. The proposed corrections provide a computationally simple way to improve boundary performance. It is imperative to acknowledge the nature of the comparison between the accuracy of Chebyshev points and equally spaced points, as highlighted in Tables 1 and 3. Our study has shown that Chebyshev points generally yield superior accuracy, particularly for certain functions and smaller values of N , we recognize that the performance may vary depending on the specific characteristics of the functions and the magnitude of N . This understanding underscores the importance of further research to explore the factors influencing the performance of different grid point distributions, ultimately guiding the selection of the most suitable approach for specific applications.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The author extends heartfelt gratitude to the University of Leicester for providing the academic environment and resources for completing his PhD during the obstacle time COVID19. Additionally, sincere appreciation goes to his supervisors, Ruslan Davidchack and Jeremy Levesley, for their invaluable guidance and mentorship. Special thanks are also extended to Taibah University for their generous financial support throughout his doctoral studies.

Conflicts of Interest: The author declare no conflict of interest.

References

1. Wendland, H. *Scattered Data Approximation*; Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press: Cambridge, UK, 2004.
2. Buhmann, M. Radial Basis Functions: Theory and Implementations. *Radial Basis Funct.* **2003**, *12*. [[CrossRef](#)]
3. Larsson, E.; Fornberg, B. A numerical study of some radial basis function based solution methods for elliptic PDEs. *Comput. Math. Appl.* **2003**, *46*, 891–902. [[CrossRef](#)]
4. Alexandridis, A.; Chondrodima, E. A medical diagnostic tool based on radial basis function classifiers and evolutionary simulated annealing. *J. Biomed. Inform.* **2014**, *49*, 61–72. [[CrossRef](#)] [[PubMed](#)]
5. Saastamoinen, A.; Lehtokangas, M.; Värri, A.; Saarinen, J. Biomedical applications of radial basis function networks. In *Radial Basis Function Networks 2: New Advances in Design*; Physica: Heidelberg, Germany, 2001; pp. 215–268. [[CrossRef](#)]
6. Shcherbakov, V.; Larsson, E. Radial basis function partition of unity methods for pricing vanilla basket options. *Comput. Math. Appl.* **2016**, *71*, 185–200. [[CrossRef](#)]
7. Wendland, H. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Adv. Comput. Math.* **1995**, *4*, 389–396. [[CrossRef](#)]
8. Wu, Z.M. Compactly supported positive definite radial functions. *Adv. Comput. Math.* **1995**, *4*, 283–292. [[CrossRef](#)]
9. Beatson, R.; Powell, M. Univariate multiquadric approximation: quasi-interpolation to scattered data. *Constr. Approx.* **1992**, *8*, 275–288. [[CrossRef](#)]
10. Wu, Z.; Schaback, R. Shape preserving properties and convergence of univariate multiquadric quasi-interpolation. *Acta Math. Appl. Sin.* **1994**, *10*, 441–446. [[CrossRef](#)]
11. Dagnino, C.; Lamberti, P.; Remogna, S. On spline quasi-interpolation through dimensions. *Ann. Dell'Universita' Di Ferrara* **2022**, *68*, 397–415. [[CrossRef](#)]
12. Maz'ya, V.; Schmidt, G. On approximate approximations using Gaussian kernels. *IMA J. Numer. Anal.* **1996**, *16*, 13–29. [[CrossRef](#)]
13. Muller, F.; Varnhorn, W. Error estimates for approximate approximations with gaussian kernels on compact intervals. *J. Approx. Theory* **2007**, *145*, 171–181. [[CrossRef](#)]
14. Cheney, E.; Light, W. *A Course in Approximation Theory*; Graduate Studies in Mathematics; American Mathematical Society: Providence, RI, USA, 2009.
15. Mason, J.C.; Handscomb, D. *Chebyshev Polynomials*; Chapman & Hall/CRC: New York, NY, USA, 2003.

16. Franz, T.; Wendland, H. Multilevel quasi-interpolation. *IMA J. Numer. Anal.* **2023**, *43*, 2934–2964. [[CrossRef](#)]
17. Ern, A.; Guermond, J.L.; Ern, A.; Guermond, J.L. Commuting quasi-interpolation. In *Finite Elements I: Approximation and Interpolation*; Springer: Cham, Switzerland, 2021; pp. 287–300.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.